NOW AVAILABLE The 512GB Mac Studio is on sale now, shipping February to March. See every configuration.

THE AI TOOLCHAIN

NO. 090

Tail

THE DAILY RELEASE FIREHOSE

PUBLISHED NOVEMBER 12, 2026 · EVERY DAY

EDITIONS **tail** | grep | head | ls | diff | uniq | perf

The daily firehose — everything the toolchain shipped today, already filtered.

▸ // HOW THIS ISSUE IS MADE SHOW

VIEW

ISSUE VIEW

scroll



SCROLL | EXPAND IN PLACE | PREVIEW PANE

INFORMATION DENSITY

everything



BRIEFED | EVERYTHING

Every tool in the issue, in full.

all 279 changes

Do you prefer **scroll**? 👍 👎

▸ // CONTENTS SHOW 67 of 68 tools

▾ // THE BRIEF

HIDE SAME ISSUE, SAME PROMPT, TWO WRITERS: GPT-5.5 | Opus 5

What stands out across today's releases, grouped by what it lets you do. Every tool named links to its entry below.

vim ? help

Three unrelated things stand out today: Sentence Transformers makes ColBERT-style retrieval cheaper to tune by controlling vector budgets; PydanticAI can keep agent work durable across Temporal/DBOS sub-agent delegations; Modal can turn a container image into an autoscaling server without writing deployment code.

BUILD

Control multi-vector retrieval quality without blowing up document storage

The release adds pooling-aware loss wrappers and hierarchical pooling controls for ColBERT-style models, where each document can produce many token vectors instead of one embedding. Teams can now tune the retrieval-quality/vector-budget tradeoff inside the library rather than hand-rolling pooling logic and hoping ranking quality survives.

Sentence Transformers

BUILD

Let agents pause, ask, delegate, and resume without losing the run

PydanticAI added a ConversationStore, mid-run user approvals via RequestInput, fallback when output is truncated, and durable Temporal/DBOS workflow support for sub-agent delegations. That removes a lot of custom state plumbing around human-in-the-loop agents and makes long-running delegated work less dependent on one process staying alive.

PydanticAI

DEPLOY

Deploy an autoscaling AI service from an existing image

Modal added no-code deployment of autoscaling servers directly from container images, plus CLI commands for managing Images. If a team already has the runtime packaged, it can expose it as a scaling service without first translating the app into a platform-specific deployment wrapper.

Modal

GOVERN

Put install integrity and account boundaries in front of agent access

magpie now verifies Ed25519 signatures on release installs, reducing the chance that an agent gateway is upgraded from a tampered artifact. ZeroClaw added OIDC multi-account dashboard sign-in with scoped per-

> account chat, so shared agent access can be separated by account instead of relying on one broad session.
>
> magpie · ZeroClaw

Does **GPT-5.5** read better? 👍 👎

BUILD

◆ AI Coding Agents

Augment Code

Feedback ⓘ changelog · Nov 12, 2026 CHANGELOG ↗

RANK

Augment Code is an AI coding assistant for VSCode and JetBrains IDEs that provides code review and context-aware suggestions, with Cosmos as its enterprise platform.

> Augment Code shipped Auggie CLI 1.2.0 with a rewritten agent, a new terminal UI, and a batch of session-management slash commands, alongside several breaking changes to tool names, keybindings, and bash output handling, plus IntelliJ plugin updates for the 2026.3 platform.

└──▷ WHAT SHIPPED · 12 FEATURES most completely described first ? what's the number?

01 Version pinning and auto-upgrade to Auggie CLI 1.x

BREAKING

90

Running `npm install -g @augmentcode/auggie` now installs the 1.x line, and existing 0.x installs with auto-update enabled upgrade automatically to 1.x. To stay on 0.x, install `@augmentcode/auggie@v0-release` and set `"autoUpdate": false` in `~/.augment/settings.json`, or set the environment variable `AUGMENT_DISABLE_AUTO_UPDATE=1`.

Pin an existing 0.x install to stay on the 0.x release line and prevent automatic upgrade to 1.x.

```
$ npm install -g @augmentcode/auggie@v0-release
```

Prevent auto-upgrade to 1.x for a 0.x install by disabling auto-update in the Augment settings file.

```json
{
  "autoUpdate": false
}
```

— Exact commands, env var, and config file given for migration.

changelog-20261112-1a151a4e

02 Agent tool renaming and bash output handling

BREAKING

85

Tool names are now `read`, `edit`, `write`, `grep`, `find`, and `bash` (hooks using old names still match for now). `bash` hook payloads no longer list changed files — use `git status` or `git diff` instead. The agent now sees up to the last 2000 lines or 50 KB of bash output plus a path to the full output, replacing the previous ~100-line excerpt.

— Exact tool names and output limits given; no migration timeline for old names.

changelog-20261112-1a151a4e

---

03 **Reasoning-effort control replaces plan-mode toggle** BREAKING

75

New `/effort` and `/effort <level>` commands (or Shift+Tab and Tab in `/model`) set the reasoning-effort level for the agent, repurposing the Shift+Tab shortcut that previously toggled plan mode.

— Named commands and shortcut change but no effort-level values listed.

changelog-20261112-1a151a4e

---

04 **New slash commands for sessions, worktrees, and plugins** NEW

68

Adds `/tree`, `/worktree`, `/export`, `/import`, and `/compact` to manage conversation branches, git worktrees, and session state; `/btw` to ask a one-shot side question while the main agent keeps running; and `/plugins` to inspect plugin commands, agents, rules, MCP servers, hooks, and skills, including disabled plugins.

— Commands named but each described in one line only. changelog-20261112-1a151a4e

---

05 **Option/Alt+Enter rebound to queue follow-up message** BREAKING

60

Option/Alt+Enter now queues a message as a follow-up instead of inserting a new line; use Shift+Enter or Ctrl+J to insert a new line instead.

— Clear before/after behavior and replacement shortcuts given.

changelog-20261112-1a151a4e

THINNER COVERAGE BELOW

06 **Per-session MCP tool discovery control** NEW

55

In `/settings`, choose Automatic, Always on, or Always off to control MCP tool discovery on a per-session basis.

— Names the setting location and options, no mechanism detail.

changelog-20261112-1a151a4e

---

07 **`/persona` removed in favor of custom agents** BREAKING

50

`/persona` is removed; use custom agents, rules, and `/permissions` instead.

— States replacement path but no setup detail for custom agents.

changelog-20261112-1a151a4e

08 **Errors tab in daemon dashboard** NEW 50

A new Errors tab in the daemon dashboard shows every retained error with a count, with independent scrolling for both Logs and Errors.

— Names UI location and behavior, no further navigation detail.

changelog-20261112-1a151a4e

09 **IntelliJ conversation history counting and retention change** IMPROVED 50

The IntelliJ plugin's conversation history now counts message content rather than request metadata, uses a larger default limit, and drops older history in larger stable steps to keep context consistent.

— Explains mechanism shift but omits concrete limit numbers. changelog-20261112-1a151a4e

10 **Rewritten terminal UI with new renderer** IMPROVED 45

Auggie CLI 1.2.0 ships a rewritten terminal UI with a new renderer for smooth scrolling and streaming, and type-to-filter support in every menu.

— Describes the change but gives no measurable specifics. changelog-20261112-1a151a4e

11 **IntelliJ plugin support for 2026.3 platform** NEW 40

Augment Code's IntelliJ plugin now supports JetBrains IDEs built on the 2026.3 platform (build 263).

— Names exact build but no further detail. changelog-20261112-1a151a4e

12 **Inline completions and Gist session sharing removed** DEPRECATED 30

Inline completions while typing and Gist-based session sharing are removed from Auggie CLI.

— Only states removal, no replacement or reasoning given. changelog-20261112-1a151a4e

└──▷ BREAKING ON UPGRADE

! `/persona` is removed; use custom agents, rules, and `/permissions` instead.

! Shift+Tab now changes reasoning effort instead of toggling plan mode.

! Option/Alt+Enter now queues a message as a follow-up instead of inserting a new line; use Shift+Enter or Ctrl+J instead.

- ! Tool names are now `read`, `edit`, `write`, `grep`, `find`, and `bash`; hooks using old names still match for now.
- ! `bash` hook payloads no longer list changed files; use `git status` or `git diff` instead.
- ! The agent now sees up to the last 2000 lines or 50 KB of bash output plus a path to the full output, replacing the previous ~100-line excerpt.
- ! Inline completions while typing and Gist-based session sharing are removed.
- ! `npm install -g @augmentcode/auggie` now installs 1.x; 0.x installs with auto-update enabled upgrade automatically to 1.x.

WAS THIS USEFUL?  

yetone magpie

Feedback ⓘ v0.1.3864 CODE ↗ RANK

Magpie is a macOS menu-bar app, TUI, and CLI that lists every AI coding agent on the machine and switches each one's model, such as Codex on DeepSeek or Claude Code on Kimi, by rewriting only that key in the agent's own config.

> This window's headline is Ed25519 signature verification on release installs, alongside Claude Code session persistence across gateway restarts and Mistral prompt-cache routing; magpie also tightened Codex's model picker, image-generation routing, and several smaller agent-bridge and display fixes.

└──▷ **WHAT SHIPPED · 11 FEATURES** most completely described first ? what's the number?

01 Claude Code session persistence across restarts NEW

90

Idle Claude Code sessions now survive a gateway restart: on shutdown, `abortAll` calls keepShelf() to write surviving sessions to `claude-shelf.json`, and the next gateway reads that file and starts Claude Code with `--resume` on the saved session, sending only the new turn instead of replaying the full conversation history.

— Full before/after mechanism with file and flag named v0.1.3860

02 Per-provider account reorder tracking BREAKING

85

Account reorder tracking now uses NoteReordered(id string) and ReorderedAt(id string) scoped per provider, so dragging accounts in one provider's pool no longer forces re-routing of conversations on other providers. This is a breaking change: NoteReordered() now requires a provider ID argument, and any caller passing no argument will not compile.

— Named breaking API change with exact signatures v0.1.3842

03 Ed25519 signature verification on updates NEW

80

magpie now checks an Ed25519 signature over `SHA256SUMS` before installing any release, so a compromised feed, proxy, or mirror cannot substitute a binary. The signed public key `3XuYQEohIRDF+zxGQry1iXGZ1cfKYFX3vsST6GSSbYQ=` is embedded in the binary, and all releases from `0.1.3862` onward must carry a valid `SHA256SUMS.sig` asset to be accepted.

— Exact key, file names and version cutoff given v0.1.3863

04 Codex image calls routed through account pool IMPROVED

80

Codex image calls to `/images/generations` and `/images/edits` now go through the account pool via `codexDrawsOnPool`, skipping accounts whose image allowance is

exhausted and signing each request with the token of the account that actually draws.

— Named endpoints and pooling mechanism `v0.1.3838`

---

05 **agy per-session effort override for Gemini** IMPROVED

80

The gateway now respects the agy agent's per-session `/effort` or `--effort` level (low, medium, high) for Gemini models it knows, letting that choice override the effort level set in the agent's row via `agyLevelAsked`.

— Names exact flags, levels and internal tracking field `v0.1.3861`

---

06 **Mistral prompt-cache routing** NEW

70

magpie's gateway now injects a `prompt_cache_key` field into Mistral chat requests, derived from the conversation session, so that Mistral routes repeated turns to the server holding the prompt cache — each session gets a consistent key, and different sessions get different keys.

— Explains mechanism and field name, no user config `v0.1.3842` `v0.1.3839`

---

07 **Codex model list falls back to local cache** IMPROVED

70

The Codex app's model list now falls back to Codex's own cached list (`models_cache.json`) when the backend cannot be reached (firewall, proxy, Windows network restrictions), so the picker shows available models instead of only GPT beside the sign-in prompt.

— Names exact file and fallback trigger conditions `v0.1.3864`

---

08 **Claude bridge routes mid-turn sub-agent reports** IMPROVED

65

The Claude bridge now correctly routes sub-agent reports that arrive mid-turn as `system -role` or `developer -role` messages — parsed via the new `taskOrAgentMsg` regex — alongside tool results rather than misattributing them as user input.

— Names exact regex and message roles involved `v0.1.3851`

---

09 **Chinese numeral units in menu bar counts** NEW

65

Menu bar token counts now respect the Chinese units setting — displaying 万/亿 (Simplified) or 萬/億 (Traditional) instead of K/M/B when the `ChineseUnits` setting is on, matching the Usage page.

— Names exact setting and unit characters `v0.1.3858`

THINNER COVERAGE BELOW

10 **Memory timestamps shown in local timezone** IMPROVED

55

Memory search results and dream notes now display timestamps in the user's local timezone with hour and UTC offset (e.g. `2026-11-11T04:30+08:00`) instead of a bare UTC date, preventing memories from appearing on the wrong calendar day.

— Example format given but no config surface v0.1.3856

---

11 **Claude Code inline thinking and effort display** NEW

20

Claude Code runs now surface their thinking inline, and a turn's own effort is shown alongside its budget.

— Single bare sentence, no mechanism or UI path v0.1.3842

└──▷ **BREAKING ON UPGRADE**

! NoteReordered() now requires a provider ID argument: NoteReordered(id string). Any caller passing no argument will not compile.

WAS THIS USEFUL? 👍 👎

Amazon Kiro Crew

Feedback ⓘ v0.12.2 NOTES ↗ RANK

Kiro Crew is an open-source workspace for coordinating AI coding agents.

> Kiro Crew added a SageMaker endpoint option to its Decisions model card, made updates install in the background without restarting the gateway, and expanded the KAS engine's agent-spec permissions model with regex-based command allowlisting.

└──▷ **WHAT SHIPPED · 3 FEATURES** most completely described first ? what's the number?

01 **Expanded KAS engine agent-spec permissions** NEW 80

The KAS engine now reads `allowedCommands` regex patterns (`\d`, `\s`, `\w`, `{n,m}`, and Rust-style expressions), `allowedPaths`, `trustedAgents`, `mcp.json` `tools` allowlists, and `useLegacyMcpJson` from a kiro-cli agent spec.

– Lists exact config keys and regex syntax supported v0.12.2

02 **SageMaker endpoint option in Decisions model card** NEW 60

The Decisions model card now accepts a SageMaker endpoint option, configured with an endpoint name and region, and hides hosted options that the organization's ceiling refuses.

– Names the endpoint fields but no exact UI path or API call v0.12.2

THINNER COVERAGE BELOW

03 **Background updates without gateway restart** IMPROVED 40

Kiro Crew now downloads updates in the background and installs them on next quit, never restarting a running gateway mid-session.

– Clear before/after but no config or flag to control it v0.12.2

WAS THIS USEFUL? 👍 👎

zot

Feedback ⓘ v0.4.74 NOTES ↗ RANK

zot manages coding agent workflows from the command line.

> zot shipped in-place compaction for its persistent storage engine and a configurable shell argv prefix for the bash tool.

└──▷ **WHAT SHIPPED · 2 FEATURES** most completely described first ? what's the number?

01 **In-place compaction for persistent store** NEW 35

zot now supports optional in-place compaction of its persistent store to reclaim space without requiring a full rebuild.

— Describes mechanism but no flag or config name given v0.4.74

02 **Configurable shell argv prefix for bash tool** NEW 35

You can configure a shell argv prefix for the bash tool, letting you wrap every bash invocation with a custom command or interpreter.

— Explains behaviour but omits exact config key or flag name v0.4.73

WAS THIS USEFUL? 👍 👎

Alibaba OpenCodeReview

Feedback ⓘ v1.13.8 NOTES ↗

RANK

OpenCodeReview is a secure, fast, efficient, battle-tested at Alibaba's scale. Hybrid architecture code review tool: deterministic pipelines + LLM Agent, precise line-level comments, built-in multi-language ruleset (NPE, thread-safety, XSS, SQL injection), OpenAI & Anthropic compatible.

> OpenCodeReview v1.13.8 adds a GitHub Copilot CLI integration and expands diff context to both sides of a comparison, while removing the `--tools` CLI override flag.

└──▷ **WHAT SHIPPED · 4 FEATURES** most completely described first ? what's the number?

01 **Removal of `--tools` CLI override flag** BREAKING 65

The `--tools` override flag has been removed from the CLI; any pipeline or script that passes `--tools` will break on upgrade.

— Names exact flag and breakage, but no migration path v1.13.8

THINNER COVERAGE BELOW

02 **GitHub Copilot CLI integration** NEW 40

OpenCodeReview adds a GitHub Copilot CLI integration via a plugin manifest, enabling Copilot CLI to delegate reviews to OCR.

— Names the integration but no manifest fields or setup steps v1.13.8

03 **Two-sided diff context for review agent** IMPROVED 35

OpenCodeReview's diff tool now reads both sides of a comparison, giving the review agent access to the before and after states of changed files.

— Explains behaviour change but no config or flag given v1.13.8

04 **MiniMax M3.1 Flash preferred in model suggestions** IMPROVED 25

OpenCodeReview now prefers MiniMax M3.1 Flash in model suggestions.

— Bare statement of a default change, no detail on mechanism v1.13.8

└──▷ **BREAKING ON UPGRADE**

! The `--tools` override flag has been removed from the CLI; any pipeline or script that passes `--tools` will break on upgrade.

WAS THIS USEFUL? 👍 👎

mvschwarz OpenRig

Feedback ⓘ v0.7.12 NOTES ↗ RANK

OpenRig is a network manager for AI coding agents: build persistent teams with roles, shared context and owned work.

> OpenRig's v0.7.12 release adds manual override commands for host pairing and permission decisions, a pull-mode relay option, Pi-RPC harness support, a herdr-app desktop terminal provider, and tightens `rig down`'s session-termination scope as a breaking change.

└──▷ **WHAT SHIPPED · 9 FEATURES** most completely described first ? what's the number?

01 **Narrower scope for rig down session termination** BREAKING

90

`rig down` now stops only a tmux session carrying the seat's own `OPENRIG_NODE_ID` on the server and session it read; a terminal OpenRig did not create, a mismarked one, or a replaced one is left running and `rig down` exits 2 naming it. A seat joined with `rig attach` as an external CLI is listed as not stopped but does not cause an exit 2.

— Full before/after behaviour with exit codes and edge cases. v0.7.12

02 **Pull-mode relay connector** NEW

81

Switch a relay connector that cannot accept incoming connections to pull mode by setting `OPENRIG_RELAY_MODE=pull` in the relay secrets file; push remains the default.

— Names env var, config location, and default behaviour. v0.7.12

03 **Manual approval commands for host pairing and permissions** NEW

80

`rig host approve` or `rig host deny` let you decide a host pairing request directly, and `rig seat decide-permission <askId> --option <id> --reason '…'` lets you decide a runner's permission request yourself, instead of waiting for a receiving agent that may never answer.

Approve a runner's permission request by ID when you want to decide it yourself rather than waiting for the receiving agent.

```
$ rig seat decide-permission <askId> --option <id> --reason
"approved for test files only"
```

— Names exact commands with flags and a runnable example. v0.7.12

04 **herdr-app terminal provider for rig terminal open** NEW

75

`rig terminal open <view> --provider herdr-app` opens a Rig view inside the herdr-app desktop window instead of a standalone terminal.

Open the OpenRig TUI inside the herdr-app desktop window rather than a standalone terminal.

```
$ rig terminal open feed --provider herdr-app
```

— Named flag with runnable example command. v0.7.12

---

05 **Pi-RPC protocol support for community harnesses** NEW

70

Declare `protocol: pi-rpc` in a community harness to use the Pi-RPC runtime; presets can now carry launch wrappers, and seats can switch between Pi and another runtime.

— Names config key and new capabilities but no example. v0.7.12

---

06 **Queue scope advice and handoff tracking** IMPROVED 70

`rig queue create` now advises on an unresolved scope, `rig queue show` names a handed-off row's successor, and a proof policy can require every listed judge.

— Names both commands and the policy change, no example. v0.7.12

---

07 **Delivery evidence tracking for sends** NEW 65

Each send now records a `transport.delivery_evidence` event capturing what was admitted, what reached the input, what the screen check showed, and for which occupant.

— Names the event and its fields, no usage path given. v0.7.12

---

08 **Manual human registration in team gateway** NEW 60

`rig gateway human add` registers a person in the team gateway when the agent does not make the offer automatically.

— Names the command but no mechanism beyond that. v0.7.12

THINNER COVERAGE BELOW

09 Seat status guidance timing and refocus hook IMPROVED

55

`rig seat status` now shows when guidance was written relative to the session's recorded moments and what the seat's refocus hook recorded.

— Describes new status output without a worked example. v0.7.12

└──▷ **BREAKING ON UPGRADE**

! `rig down` now stops only a tmux session carrying the seat's own `OPENRIG_NODE_ID` on the server and session it read; a terminal OpenRig did not create, a mismarked one, or a replaced one is left running and `rig down` exits 2 naming it. A seat joined with `rig attach` as an external CLI is listed as not stopped but does not cause an exit 2.

WAS THIS USEFUL? 👍 👎

corecoder

Feedback ⓘ v0.12.0 NOTES ↗ RANK

Corecoder is a minimal AI coding agent (~1,000 lines of Python) inspired by Claude Code. Works with any LLM.

> CoreCoder's two releases both deepen its MCP support, adding the ability to read text resources from MCP servers and a new HTTP transport for connecting to hosted MCP tool servers.

└──▷ **WHAT SHIPPED · 2 FEATURES** most completely described first ? what's the number?

01 **HTTP transport for hosted MCP tool servers** NEW 88

Connect to hosted MCP tool servers by adding a `url` and optional authentication headers to `~/.corecoder/mcp.json`; local `command` entries continue working alongside remote ones. The new HTTP transport accepts JSON and SSE responses, manages session headers, and limits response bodies to 4 MiB — reusing the existing consent gate, hooks, and the `/mcp reconnect` command with no new SDK dependency.

— Names exact config file, keys, limit and reconnect command v0.11.0

02 **MCP text resource reading over stdio and HTTP** NEW 73

CoreCoder now reads text resources published by MCP servers over stdio or Streamable HTTP, using one on-demand bridge per server that covers catalog listing, URI templates and text reads — including resource-only servers with no tools. Binary resource contents and resource reads exceeding 32,768 characters now fail explicitly rather than silently truncating or corrupting output, and these reads go through the existing consent, plan-mode and hook checks with no new permission surface added.

— Names mechanism, limits and transport but no exact command shown v0.12.0

WAS THIS USEFUL? 👍 👎

OpenAI Codex CLI

Feedback ⓘ rust-v0.170.0-alpha.2 NOTES ↗

CODE ↗ RANK

OpenAI Codex CLI runs an agent in the terminal that reads, changes, and tests code in local repositories.

> Codex CLI expands provider management with in-TUI provider editing and a model/list filter, adds device-code OAuth gateway sign-in and multiple simultaneous side chats, and rolls out incremental tool-declaration updates by default on Google Cloud Vertex AI while making them opt-in for regular OpenAI Responses API calls.

└──▷ **WHAT SHIPPED · 1 FEATURE** most completely described first ? what's the number?

01 Incremental tool updates for Responses API and Vertex AI

IMPROVED 90

Codex CLI adds opt-in incremental tool updates for regular OpenAI Responses API calls, and now sends tool declarations incrementally by default when using the Google Cloud Vertex AI provider, reducing redundant full-catalog resends on each request. Opt out of the Vertex AI default by setting `incremental_tools_legacy_gcp = false` under `[features]` in `config.toml`.

Disable incremental tool updates on Google Cloud Vertex AI if your setup requires the full tool catalog to be sent on every request.

```toml
[features]
incremental_tools_legacy_gcp = false
```

— Names exact config key, file and section plus behaviour change rust-v0.167.2 rust-v0.170.0-alpha.2

THINNER COVERAGE BELOW

WAS THIS USEFUL? 👍 👎

Letta Code

Feedback ⓘ v0.34.54 NOTES ↗ RANK

Letta Code is a terminal coding agent with persistent memory and identity across development sessions.

> Letta Code tightened sandbox security by making app exposure private by default, and expanded the context available to shell commands and external tool integrations.

└──▷ **WHAT SHIPPED · 3 FEATURES** most completely described first ? what's the number?

01 **Sandbox app exposure private by default** BREAKING 42

Sandbox app exposure is now private by default, reducing the attack surface of apps spun up during agent execution.

— Describes default change but no toggle or flag named v0.34.52

02 **Turn input time forwarded to external tool calls** NEW 31

External tool calls now receive the turn input time, giving downstream integrations accurate timing context for each agent turn.

— No field name or usage example provided v0.34.54

03 **Readable agent values in shell commands** NEW 29

Shell commands now have access to readable agent values as variables, expanding what can be passed to external processes at execution time.

— No variable names or syntax given v0.34.54

WAS THIS USEFUL? 👍 👎

zentty

Feedback ⓘ v0.6.1 NOTES ↗ RANK

Zentty is a native macOS terminal for agent-driven development, built on Ghostty.

> Zentty v0.6.1 adds GitLab merge request visibility to its sidebar and improves the Agent History experience when history tracking is disabled.

└──▷ **WHAT SHIPPED · 2 FEATURES** most completely described first ? what's the number?

01 **Agent History disabled-state screen** IMPROVED 40

Agent History now shows a screen with a shortcut to Settings when history is turned off.

— Describes behaviour and a clear navigation path v0.6.1

02 **GitLab merge requests in sidebar** NEW 35

Zentty now shows GitLab merge requests in the sidebar.

— Names the surface but no detail on scope or mechanism v0.6.1

WAS THIS USEFUL? 👍 👎

Zed

Feedback ⓘ v1.28.1 NOTES ↗ RANK

Zed is an open-source code editor that runs collaborative development workflows.

> Zed's v1.28.1 release adds automatic Git fetching with configurable polling, a resizable Refs column and other Git workflow UX tweaks, plus new settings for debugger panel sizing, edit prediction button visibility, and mouse-wheel font zoom in diffs and terminals.

└──▷ **WHAT SHIPPED · 5 FEATURES** most completely described first ? what's the number?

01 Automatic Git fetching for current and default branches NEW

88

`git.auto_fetch.enabled` enables automatic fetching of the current branch's upstream remote and the default branch's remote; the polling interval is set via `git.auto_fetch.interval_secs`, which defaults to 60 seconds.

Keep your local Git refs up to date automatically without manually running fetch, checking every 2 minutes instead of the default 60 seconds.

json

```
{
  "git": {
    "auto_fetch": {
      "enabled": true,
      "interval_secs": 120
    }
  }
}
```

— Exact config keys, default value, and a runnable example. v1.28.1

02 Mouse-wheel font zoom in diffs and terminals NEW

78

When `mouse_wheel_zoom` is enabled, Zed supports zooming font size via the mouse wheel in diff views and terminals, triggered with `cmd-scroll` on macOS and `ctrl-scroll` on Linux/Windows.

— Names the setting and exact key combinations, no example given. v1.28.1

03 Configurable default debugger panel size NEW

75

`debugger.default_width` and `debugger.default_height` settings let you fix the debugger panel's starting dimensions on launch.

Set a fixed starting size for the debugger panel so it occupies a predictable portion of the screen on launch.

```json
{
  "debugger": {
    "default_width": 480,
    "default_height": 300
  }
}
```

— Named config keys with a runnable example, minimal further mechanism. v1.28.1

THINNER COVERAGE BELOW

04 **Git workflow navigation and interaction improvements** NEW

57

Zed adds a resizable Refs column to the Git Graph, toggleable on or off from the column header's context menu; supports dragging files from the Git Panel into terminal and editor panes; and adds branch-name search in the worktree picker.

— UI paths described but no config keys or commands. v1.28.1

05 **Toggle for edit prediction status bar button** NEW 50

The `edit_predictions.button` setting toggles visibility of the edit prediction button in the status bar.

— Names the config key but no example or further behavior. v1.28.1

WAS THIS USEFUL? 👍 👎

All Hands AI OpenHands

Feedback ⓘ v1.39.0 NOTES ↗

RANK

OpenHands is an open-source software-development agent that writes code, runs commands, and browses the web.

> OpenHands' latest release adds filtering controls to the automations dashboard and links Canvas families to Agent Server verification.

└──▷ **WHAT SHIPPED · 2 FEATURES** most completely described first ? what's the number?

01 **Service Accounts filter in automations dashboard** NEW

42

The automations dashboard gains a new Service Accounts owner filter option, allowing users to separate service-owned automations from user-owned ones.

— Names the UI filter but no further mechanism v1.39.0

02 **Canvas families linked to Agent Server features** IMPROVED

23

Canvas families are now linked to Agent Server features in the server-side verification layer.

— Bare statement with no detail on mechanism or use v1.39.0

WAS THIS USEFUL? 👍 👎

zerostack

Feedback ⓘ v1.8.6 NOTES ↗ RANK

zerostack is a lightweight coding agent written in Rust, optimized for memory footprint and performance

> zerostack v1.8.6 disables `deny_repeated_reads` by default, adds Pijul version control support, and ships a Nix Home Manager module for declarative configuration.

└──▷ **WHAT SHIPPED · 3 FEATURES** most completely described first ? what's the number?

01 Nix Home Manager module NEW

81

A new Home Manager module at `nix/module/home-manager.nix` lets users manage zerostack declaratively, configured via `programs.zerostack.enable`, `programs.zerostack.settings`, and `programs.zerostack.prompts`, with settings expressed as Nix (mirroring TOML) and prompts as Markdown strings or derivation paths.

Manage zerostack declaratively via Home Manager, with settings expressed as Nix and prompts as Markdown strings or derivation paths.

nix

```nix
imports = [
  (import "${inputs.zerostack}/nix/module/home-manager.nix")
];

programs.zerostack = {
  enable = true;
  settings = { /* TOML in the form of Nix */ };
  prompts = { /* keys mapping to derivation paths or Markdown
strings */ };
};
```

— Names exact file path and config keys with a usage example v1.8.6

02 `deny_repeated_reads` off by default BREAKING

65

zerostack now defaults `deny_repeated_reads` to off, reducing friction for agents that read the same file more than once in a session. Installations that relied on it being on must explicitly re-enable it in config.

— Names config key and gives before/after plus migration note v1.8.6

THINNER COVERAGE BELOW

03 Pijul version control support NEW

25

Adds Pijul version control commands, extending zerostack's VCS support beyond Git.

— No commands or mechanism named beyond prose `v1.8.6`

└──▷ **BREAKING ON UPGRADE**

! `deny_repeated_reads` is now off by default; installations that relied on it being on must explicitly re-enable it in config.

WAS THIS USEFUL? 👍 👎

ogcode

Feedback ⓘ v0.59.0 NOTES ↗ RANK

Ogcode is an open-source, local-first AI teammate and agent operating environment for turning intent into completed work—with tools, memory, permissions, and human oversight.

> Ogcode v0.59.0 adds a machine-wide background model setting for internal tasks, per-request task-identity headers for model routing, and a consolidated page-by-page document reader, alongside CLI and deep-search reliability fixes.

└──▷ **WHAT SHIPPED · 5 FEATURES** most completely described first ? what's the number?

01 **Background model setting for internal tasks** NEW 95

A single background model can now be configured for titles, summaries, compaction, risk assessment, intent matching, indexing, and `task` sub-agents via Settings → General → Background tasks, backed by `GET` / `POST /api/models/background`. The default is Inherit from active session; a pinned model that is later removed falls back to inheriting rather than failing, and a model not currently served is refused at save time rather than at first use.

Pin a dedicated background model so all summaries, compaction, and risk checks use it without touching the user-facing session model.

```
$ curl -X POST /api/models/background \
  -H 'Content-Type: application/json' \
  -d '{"model": "gpt-4o-mini"}'
```

Inspect what the background model endpoint currently returns before pinning a new model.

```
$ curl GET /api/models/background
```

— Endpoint, UI path, defaults, fallback and runnable curl examples given v0.59.0

02 **doc_index replaces docx_index and pdf_index** BREAKING

80

New `doc_index` tool replaces the listing-only `docx_index` and `pdf_index`, returning a page catalogue a step can choose from; `read_docx_page` and `read_pdf_page` now accept an `end_page` range and mark each page with `--- page N ---`, so only the pages actually needed are read. Callers using the old `docx_index` or `pdf_index` tools must switch to `doc_index`.

— Names old and new tools and the page-range mechanism v0.59.0

03 **Task-identity headers for model routing** NEW

75

Every provider request now carries X-Client-Task naming one of fourteen call kinds (turn, sub-agent, indexing, note, search, recall, intent matching, turn memory, compaction, title, risk gate, rewrite, probe, credential validation), plus X-Client-Complexity (easy, medium, or hard) and X-Client-Workload (read-heavy or write-heavy) on delegated children, so a router can direct cheaper models to lighter work. The `task` and `teammate` tools now accept X-Client-Complexity and X-Client-Workload as arguments and reject misspellings rather than silently sending nothing.

— Headers and values fully named but no runnable example v0.59.0

04 **ogcode index failure reporting** IMPROVED

60

The `ogcode index` command now exits non-zero when indexing fails instead of printing a false success, and reports per-document progress with a terminal phase even on failure.

— Exact command named with clear before/after behaviour v0.59.0

THINNER COVERAGE BELOW

05 **Deep search ranker budget increase** IMPROVED

55

Deep search ranker output budget raised from 256 to 1024 tokens, and an empty answer caused by hitting the budget now falls back to the pages found rather than parsing unfinished reasoning for ranking digits.

— Concrete numbers given but no user-facing action v0.59.0

└──▷ **BREAKING ON UPGRADE**

! `doc_index` replaces `docx_index` and `pdf_index`; callers using either of the old tools must switch to `doc_index`.

WAS THIS USEFUL? 👍 👎

pstack-claude

Feedback ⓘ v0.9.136 NOTES ↗ RANK

pstack-claude is a port of Poteto's pstack agent framework to Claude Code, Codex, Copilot, Pi, and other AI coding assistants.

> pstack-claude adds home-directory playbooks that apply across every repository without copying them in, along with validation and ordering support for the new location.

└──▷ **WHAT SHIPPED · 1 FEATURE** most completely described first ? what's the number?

01 **Home-directory playbooks support** NEW 78

poteto-mode now reads playbooks from `~/.agents/playbooks/` in addition to the repository's `.agents/playbooks/`, so a single playbook can apply across all repositories without copying it into each one. `check-playbooks.mjs` checks both `~/.agents/playbooks/` and the repository's `.agents/playbooks/` folders, identifies home playbooks by their `~/.agents/playbooks/` path, and checks a repository at the home folder once. When playbooks from both the home folder and the repository folder match a task, home-folder changes apply first.

— Names exact paths and script but no command example v0.9.136

WAS THIS USEFUL? 👍 👎

Amazon Kiro

Feedback ⓘ

CLI 3.2.1: Full-Screen Diagram Viewer and Sign-In to More MCP Servers (2026-11-11)

BLOG ↗ RANK

Kiro is an agentic development environment that uses spec-driven workflows to plan, build, and maintain software.

> Kiro's CLI picked up a full-screen Mermaid diagram viewer, configurable tool-call output detail, and Identity Center/headless login, while Kiro Crew (formerly Mates, now Bots) added browser automation with live-view takeover, an Amazon Bedrock Knowledge Base integration, new Strands and GitHub Copilot CLI harnesses, a conductor/worker mates preview, and made Kiro Agent the default harness in place of a bundled Kiro CLI.

└──▷ **WHAT SHIPPED · 25 FEATURES** most completely described first ? what's the number?

01 New `kirocrew` CLI subcommands for spawning, agents, and version removal

NEW 85

Run `kirocrew spawn run` with `--model`, `--agent`, and `--crew` flags to control which model and agent a spawned run uses. Run `kirocrew agent create` to create a new agent from the command line. Run `kirocrew kas remove` with `--version` to remove a specific installed Kiro Agent version (a blank `--version` now refuses instead of removing all versions).

Spawn a run targeting a specific model, agent, and crew from the command line to automate a recurring engineering task.

```
$ kirocrew spawn run --model claude-sonnet-4-5 --agent my-agent --crew my-crew
```

— Three runnable subcommands with exact flags Crew: Kiro Agent by Default and Chat Yo…

02 Output details controls in `/settings output-details` NEW

80

Control how much tool call information is shown by running `/settings output-details` and choosing the `collapsed`, `lean default`, or `full` preset, or set tool arguments, reasoning, output filters, subagent detail, and truncation individually.

Tune tool call verbosity mid-session to see only the information you need without leaving the chat.

📍 `/settings` output-details

— Exact command and preset names given CLI: Output Details and Sign-In Fixes (…

03 **CLI login for Identity Center and headless Linux** NEW

80

Sign in to an Identity Center instance from the CLI by running `kiro-cli login --license pro` with `--identity-provider` and `--region`. On Linux hosts with no graphical session, `kiro-cli login` at a terminal now uses the device flow and prints a code and URL instead of attempting a browser-based sign-in.

Authenticate to a specific AWS Identity Center instance from a headless CI or server environment.

```
$ kiro-cli login --license pro --identity-provider <provider-url>
--region <aws-region>
```

— Runnable command with flags and fallback behavior

CLI: Output Details and Sign-In Fixes (…

04 **Browser automation live-view and takeover** NEW

80

Pick which browser the agent drives and control the live view, its quality, frame rate, and recording via Settings → Browser (replacing the former built-in browser switch and extension token field); when the agent uses the browser the page streams live into chat so you can click to take over and hand it back when done, with a secret-entry sheet so credentials are never seen by the agent.

Let the agent handle a login flow without exposing the password: watch the browser live and hand credentials over via the secret-entry sheet.

📍 In Kiro Crew, go to Settings -> Browser, choose your preferred browser (Chrome, Edge on Windows, or Kiro's own Chromium) and turn the live view on. When the agent reaches a sign-in page, the chat shows the page live; click the picture to take over or use the secret-entry sheet to type credentials — nothing typed is saved and the agent never sees it.

— Mechanism, scope, and settings path all given Crew: Bots for Everyone and a Browser Y…

05 **Full-screen Mermaid diagram viewer in CLI chat** NEW

75

Open wide Mermaid diagrams in a full-screen viewer with `Ctrl+]` or the `/diagram` command; use arrow keys or mouse wheel to scroll, Tab to step through older

diagrams, and `Esc` to return to chat. The key used to open it can be changed via `chat.keybindings.viewDiagram` in configuration.

When an agent reply contains a Mermaid diagram too wide for your terminal, open it at full width to scroll and inspect it.



— Names exact command, keys and config key `CLI 3.2.1: Full-Screen Diagram Viewer a…`

---

06 **Chat settings: auto-compact threshold, jump-to-bottom, copy-on-select**

IMPROVED 75

Fine-tune the auto-compact threshold to any exact percent from 5 to 90 under `Settings → Chat → Advanced`, disable the jump-to-bottom-on-send behavior via `Settings → Chat → Jump to the bottom when you send`, and enable copy-on-select for chat transcripts via the opt-in `Settings → Chat → Transcript → Copy on select`.

— Each toggle's exact settings path is named `Crew: Bots for Everyone and a Browser Y…`

---

07 **New feature previews: Ultra Mode, Explore, per-Bot harness, Shared Crew, Remote dispatch, and Mates**

NEW 75

Adds off-by-default preview switches under `Settings → Developer → Feature Previews` for Ultra Mode (renamed from Crew Mode, with a redesigned Board), Explore, Agent harness per Bot, Shared Crew, Remote dispatch, and Mates. Shared Crew lets the owner share their machine over Tailscale via a two-step card under `Settings → Developer → Feature Previews → Shared Crew` after a restart. Mates, enabled via `Settings > Developer > Feature Previews > Mates`, lets mates act as conductors with their own worker sessions, each mate getting an agent file under `~/.kiro/mates/agents/`, importable from a `.crewmate` file for portability across installs.

Enable the conductor-and-worker mates preview so each mate runs hands-on work in a dedicated worker session.

> 📍 1. Open Settings › Developer › Feature Previews 2. Enable `Mates` 3. Existing mates are migrated to ~/.kiro/mates/agents/ automatically on next startup

— Names all five previews and the Tailscale mechanism Named settings path, file location and import format

`Crew: Bots for Everyone and a Browser Y…` `Crew: Kiro Agent by Default and Chat Yo…`

---

08 **Per-job schedule limits and cron strictness controls** NEW

75

Configure per-job schedule limits using the Schedule form: a Time limit (default 30 minutes) and an 'Auto-pause after N failures in a row' count, where 0 means never pause. Set `strict_schedule` on an app's cron to fire exactly on schedule with no spread. Run `kirocrew cron add update` with `--auto-pause-after` to set the same failure-backoff limit from the command line.

Keep remote MCP sign-ins alive across sessions so you are not prompted to re-authenticate on every new Kiro Agent session.

```
$ kirocrew cron add update --auto-pause-after 3
```

— Form fields, config key and CLI flag all named Crew: Kiro Agent by Default and Chat Yo…

09 **Kiro Agent now the default harness; Kiro CLI no longer bundled**

75

BREAKING

Kiro Agent is now the default harness; a new install with no saved harness choice runs new sessions on Kiro Agent, while a previously chosen harness, including Kiro CLI, is preserved. The desktop app no longer bundles Kiro CLI — it ships the Kiro Agent harness it was tested with and detects a self-installed Kiro CLI. The toggle between Kiro Agent and Kiro CLI now lives in `Settings > Developer > Feature Previews` instead of its previous location.

— Clear before/after and relocation of toggle Crew: Kiro Agent by Default and Chat Yo…

10 **Amazon Bedrock Knowledge Base in Crew capabilities** NEW

70

Register an Amazon Bedrock Knowledge Base under Capabilities → Knowledge → Sources → Add Knowledge Base to query it live and merge results into knowledge search, with a cost-confirmation card shown before the first query.

Attach an Amazon Bedrock Knowledge Base so the agent can search your internal docs before the first query commits any cost.

> 📍 In Kiro Crew, go to Capabilities -> Knowledge -> Sources, click Add Knowledge Base, select your existing Amazon Bedrock Knowledge Base, confirm the AWS account and Region on the cost-confirmation card, then run a chat that references internal documentation.

— Exact UI path and cost-confirmation mechanism named Crew: Bots for Everyone and a Browser Y…

11 **Push notifications and custom sounds settings** NEW

70

Turn on push notifications that fire after the app is closed via `Settings → Notifications → Push notifications → Notify me even when Kiro Crew is closed`, including iOS delivery after Add to Home Screen, and set a custom notification sound in any category via `Settings → Notifications → Custom sounds`, accepting short MP3, WAV, or OGG files.

Receive Kiro Crew approval alerts on your phone even after closing the browser or desktop app.

> 📍 In Kiro Crew, go to Settings -> Notifications -> Push notifications and enable 'Notify me even when Kiro Crew is closed'. On iOS, also tap Share -> Add to Home Screen first so the web app can receive alerts.

— Exact settings path and supported audio formats

`Crew: Bots for Everyone and a Browser Y…`

---

12 **Strands and GitHub Copilot CLI agent harnesses** NEW

70

Kiro Crew adds Strands and GitHub Copilot CLI as agent harness choices under `Settings → Agent Harness` (Strands requires the Strands CLI installed, with model provider keys stored in `Settings → Secrets`), and every harness's model picker now shows that backend's own model names.

— Named harnesses and settings paths, no command `Crew: Bots for Everyone and a Browser Y…`

---

13 **MCP sign-in improvements** IMPROVED

65

Kiro now signs in automatically to MCP servers that support Client ID Metadata Documents but offer no dynamic client registration (such as Tableau's hosted MCP) using its own hosted client metadata document, so no `oauth` block is needed in configuration, and setting `agent.kas_persist_mcp_sign_in` keeps remote MCP sign-ins persistent across sessions without re-authenticating each time (off by default in 0.11.0, on by default in 0.11.2).

— Mechanism named but no command to try it Named config key with default-change history

`CLI 3.2.1: Full-Screen Diagram Viewer a…` `Crew: Kiro Agent by Default and Chat Yo…`

---

14 **Consistent tool call output formatting in CLI chat** IMPROVED

60

Tool calls now share a consistent look: shell, search, and generic output folds under the tool call corner, finished calls show returned file lines, two or more consecutive finished tool calls collapse into one summary row, file-write approvals show a full diff instead of a capped preview, and finished searches keep their result count in scrollback.

— Detailed behavior change but no command to invoke

`CLI: Output Details and Sign-In Fixes (…`

THINNER COVERAGE BELOW

15 **Bot profile export and import as `.kirobot` files** NEW

55

Export a Bot's profile as a `.kirobot` file via Export as file on its profile card, and re-import it via Bots → Import from file.

— Named file format and UI path, thin description

Crew: Bots for Everyone and a Browser Y…

16 **Workflow monitor iteration-cap and autonomous-mode notices**

IMPROVED

50

Workflow loops that pause after reaching their iteration cap now offer two choices in the workflow monitor: run one more iteration, or mark the loop complete and continue the run. Kiro also now tells you when a workflow is requested in autonomous mode instead of silently passing the request on.

— Describes behavior but no exact UI path CLI 3.2.1: Full-Screen Diagram Viewer a…

17 **Pop chat session into its own window** NEW

50

Pop the current session into its own window with Cmd+O (or Ctrl+O on Windows and Linux).

— Exact shortcut given but nothing more Crew: Bots for Everyone and a Browser Y…

18 **Bitbucket Server hosts for pull-request sources** NEW

50

Add Bitbucket Server hosts for pull-request sources via the `dashboard.bitbucket_server_hosts` setting.

— Single config key, no further detail given Crew: Kiro Agent by Default and Chat Yo…

19 **Developer Mode removed from Settings** BREAKING

50

Developer Mode is removed; Settings → Developer now always shows Feature Previews, Logs, System, Telemetry, Storage, Memory, Config, Debug tools, and Archive.

— Lists new always-visible sections exactly Crew: Bots for Everyone and a Browser Y…

20 **Usage tab now shows credits, not estimated tokens** BREAKING

50

The Usage tab no longer shows estimated tokens; the Estimated Tokens card and 'Est. tokens' column are removed. The tab now shows credits reported by the provider.

— Names removed UI elements and replacement metric

Crew: Kiro Agent by Default and Chat Yo…

21 **Color-blind safe and Amethyst OLED themes** NEW 45

Apply Color-blind safe or Amethyst OLED themes via Settings → Display → Theme.

— Single-line addition, named themes and path Crew: Bots for Everyone and a Browser Y…

22 **Session filtering and cleanup window options** NEW 45

Filter and group sessions with new Group by kind and Not in progress options in the sidebar Filter menu, and clean up sessions with new 14-day and 30-day windows.

— Thin list of UI options, no deeper mechanism Crew: Bots for Everyone and a Browser Y…

23 **Mates renamed to Bots** BREAKING 40

Mates are renamed to Bots across every screen, prompt, and command bar row; older `.crewjam` and `.crewmate` export files still import.

— Names old file formats but no migration steps Crew: Bots for Everyone and a Browser Y…

24 **MCP Management moved to Connections page** BREAKING 40

MCP Management moved from its previous location to the Connections page, below the MCP Servers list.

— Thin one-line relocation note Crew: Bots for Everyone and a Browser Y…

25 **Summon hotkey off by default on new installs** BREAKING 40

The summon hotkey is off on new installs; turn it on in Settings. Existing installs keep their setting.

— Minimal detail beyond the default change Crew: Kiro Agent by Default and Chat Yo…

└──▷ BREAKING ON UPGRADE

! Mates are renamed to Bots across every screen, prompt, and command bar row; older `.crewjam` and `.crewmate` export files still import.

! Developer Mode is removed; Settings → Developer now always shows Feature Previews, Logs, System, Telemetry, Storage, Memory, Config, Debug tools, and Archive.

! MCP Management moved from its previous location to the Connections page, below the MCP Servers list.

! Settings → Browser replaces the former Use the built-in browser switch and the extension token field.

! Crew Mode is renamed Ultra Mode and remains an off-by-default Feature Preview with a redesigned Board.

! Kiro Agent is now the default harness; a new install with no saved harness choice runs new sessions on Kiro Agent. A previously chosen harness, including Kiro CLI, is preserved.

! The desktop app no longer bundles Kiro CLI; it ships the Kiro Agent harness it was tested with and detects a Kiro CLI you installed yourself.

! The toggle between Kiro Agent and Kiro CLI now lives in Settings > Developer > Feature Previews instead of its previous location.

! The summon hotkey is off on new installs; turn it on in Settings. Existing installs keep their setting.

! The Usage tab no longer shows estimated tokens; the `Estimated Tokens` card and 'Est. tokens' column are removed. The tab now shows credits reported by the provider.

WAS THIS USEFUL? 👍 👎

◆ Coding Agent Add-ons

Token Optimizer

Feedback ⓘ v5.13.71 NOTES ↗ RANK

Token Optimizer is a coding-agent plugin that trims wasted context, checkpoints work across sessions and compactions, and reports token spend on a live dashboard.

> Token Optimizer added in-session savings visibility via Claude Code's status line and a new Savings page, alongside secrets-free settings backup and safer tracking across parallel sessions.

└──▷ **WHAT SHIPPED · 4 FEATURES** most completely described first ? what's the number?

01 **Secrets-free settings backup with opt-out** NEW 80

Token Optimizer backs up settings without storing secrets; set `TOKEN_OPTIMIZER_SETTINGS_HEAL=0` to disable automatic restoring of settings on startup, useful when credentials are managed outside the tool and restore announcements are unwanted.

Prevent Token Optimizer from automatically restoring settings on startup — useful when you manage credentials outside the tool and don't want any restore announcements.

```
$ TOKEN_OPTIMIZER_SETTINGS_HEAL=0 claude
```

— Exact env var and runnable command given v5.13.70

02 **Savings page with measured/estimated metrics** NEW 75

A new Savings page shows last-30-day and since-install figures, tokens kept out of context, transformation cost, and usage limits as bars, with each figure labelled as measured or estimated along with its before source. Claude Code mod commands Keep warm, Clean up, and Start fresh now count toward savings with their costs booked, and transformation cost is now priced at the models actually used so the figure updates after a model upgrade in either direction.

— Names specific metrics and commands, only a UI path to act on v5.13.70

03 **Status line savings band and limits card** NEW 68

Claude Code's status line now shows token savings directly in the band, with tooltip explanations on headline numbers and one click to open the Savings page — no dashboard required. A new limits card answers how much more work fits in your weekly limit compared with your own baseline, and shows how the figure was calculated. Savings display filters by recency and threshold: this week's savings appear when above $20, otherwise the last 30 days appear when above $60 with tokens kept out of context, otherwise nothing is shown.

— Names exact dollar thresholds but only a UI path, no command v5.13.71

THINNER COVERAGE BELOW

04 **Parallel-session-safe savings tracking** IMPROVED 15

Token Optimizer's savings tracking is now safe to use across parallel sessions, per the release summary; no further mechanism or scope is described.

— Only named in summary, no mechanism or detail given v5.13.70

WAS THIS USEFUL? 👍 👎

◆ Image & Video Generation

Hugging Face Diffusers

Feedback ⓘ v0.43.0 NOTES ↗

RANK

Diffusers provides pretrained diffusion pipelines, schedulers, and models for image, video, and audio generation in PyTorch.

> Diffusers v0.43.0 adds five new modular generation pipelines spanning image, video, and speech-driven animation, an AMD RDNA4 attention backend, and several breaking changes tightening dependency, security, and training behavior.

└──▷ WHAT SHIPPED · 12 FEATURES most completely described first ? what's the number?

01 **PAN 2 generative simulator pipeline** NEW 80

`PAN 2` integrates a 29B generative simulator for text-to-video and image-to-video generation from scene and camera-movement prompts, with context parallelism, FirstBlockCache, MagCache, and TaylorSeer caching, and requires the `pan2-guardrail` package for prompt and video safety checks.

— Names caching mechanisms and required package but no exact command v0.43.0

02 **MiniMax-H3-Image modular pipeline** NEW 75

Adds `MiniMax-H3-Image` for text-to-image generation and editing using one or more reference images, loaded via ModularPipeline.from_pretrained(...) and load_components(...).

Load and run MiniMax-H3-Image for text-to-image generation using the new modular pipeline API.

```python
from diffusers import ModularPipeline

pipeline = ModularPipeline.from_pretrained('MiniMaxAI/MiniMax-H3-Image')
pipeline.load_components()

output = pipeline('A serene mountain lake at sunrise, photorealistic')
output.images[0].save('output.png')
```

— Has a runnable code example and named API calls v0.43.0

03 **Kandinsky S2V 1.0 speech-to-video pipeline** NEW 75

`Kandinsky S2V 1.0` animates a person from a reference image and a speech recording, generating video in five-second chunks using `PiflowScheduler` with a default of 16 steps and carrying FramePack memory between chunks.

— Detailed mechanism but no runnable command shown v0.43.0

04 **P-H3-Image pipeline from Pruna AI** NEW 70

P-H3-Image performs text-to-image generation and instruction-based editing with up to eight input images, including a prompt upsampler, image autoencoder, and text connector.

— Names components but no example invocation v0.43.0

05 **AMD RDNA4 attention backend** NEW 70

Adds an `aiter_sage_hub` attention backend that runs a Hub-hosted SageAttention kernel on AMD RDNA4 gfx1201 GPUs, with native SDPA fallback for unsupported input shapes.

— Names the backend and GPU target but no setup command v0.43.0

06 **Pika H3 Flex video+audio pipeline** NEW 65

Pika H3 Flex generates video and synchronized stereo audio in six steps from text prompts, first and/or last frames, and multimodal references, supports LoRA adapter loading, and can share already-loaded `MiniMax-H3` components.

— Mechanism described but no concrete usage command given v0.43.0

07 **Model component import reorganization** BREAKING 65

Latent upscalers and condition embedders move under `diffusers.models`, with public imports such as `from diffusers import LTX2TextConnectors`; old pipeline-module imports are deprecated for removal in 1.0.0.

— Gives exact new import path and deprecation timeline v0.43.0

08 **Gradient clipping fix in LoRA training scripts** BREAKING 65

`--max_grad_norm` now takes effect in the text-to-image LoRA training script `train_text_to_image_lora.py` and the SDXL LCM LoRA script `train_lcm_distill_lora_sdxl.py`; runs with gradients above the clipping threshold can behave differently from earlier versions.

— Names exact flag and affected scripts v0.43.0

09 **Dependency minimum version enforcement** BREAKING 60

Dependency version checks now enforce supported minimums — `torch>=2.6` and `transformers>=4.41.2` — treating an installed package below the minimum as unavailable and showing an upgrade message.

— Names exact version floors but no migration steps v0.43.0

THINNER COVERAGE BELOW

10 **Tensor parallelism improvements for MiniMax-H3** IMPROVED

50

Tensor parallelism for MiniMax-H3 now shards the transformer, including its large modulation projections, across devices, and replicated parameters retain their intended dtype during tensor-parallel loading.

— Describes the change but no config or command shown v0.43.0

11 **Custom component import trust checks** BREAKING

50

Custom component modules outside the supported library set must already be imported by the caller; repository configuration no longer causes arbitrary component-library imports, and modular custom-block loading checks remote-code trust before processing requirements.

— Explains before/after but no flag or config key given v0.43.0

12 **InputParam required-with-default behavior change** BREAKING

50

An `InputParam` declared both required and with a non-`None` default is now treated as optional and emits a warning; explicitly passing `None` preserves that value rather than applying the block default.

— Describes precise behavior change but no code sample v0.43.0

└──▷ BREAKING ON UPGRADE

! Old pipeline-module imports for latent upscalers and condition embedders are deprecated and will be removed in 1.0.0; use public imports such as `from diffusers import LTX2TextConnectors` instead.

! `--max_grad_norm` now takes effect in the text-to-image LoRA training scripts (`train_text_to_image_lora.py`) and SDXL LCM LoRA script (`train_lcm_distill_lora_sdxl.py`); runs with gradients above the clipping threshold can behave differently from earlier versions.

! Custom component modules outside the supported library set must already be imported by the caller; repository configuration no longer causes arbitrary component-library imports, and modular custom-block loading checks remote-code trust before processing requirements.

! An `InputParam` declared both required and with a non-`None` default is now treated as optional and emits a warning; explicitly passing `None` preserves that value rather than applying the block default.

WAS THIS USEFUL? 👍 👎

◆ Speech & Voice AI

Daily Pipecat

Feedback ⓘ v1.16.0 NOTES ↗

Pipecat is a framework for realtime voice and multimodal agents, orchestrating speech-to-text, model inference, and text-to-speech as a streaming pipeline.

Pipecat adds diarization, Zoom AI and Gemini Interactions LLM services, Sarvam HTTP STT, and new tuning fields across Cartesia, AssemblyAI, ElevenLabs, OpenAI Realtime, Inworld, xAI, Piper, MiniMax, Resemble AI, Smallest, Simli, and Genesys services.

└──▷ GET THIS VERSION ▾

```
$ git clone --branch v1.16.0 https://github.com/pipecat-ai/pipecat.git
# already have the repo? check out this version:
$ git checkout v1.16.0
```

Control concurrent ElevenLabs TTS synthesis with the new `pause_frame_processing` option on `ElevenLabsTTSService` and `ElevenLabsDialogueTTSService`; set it to `False` to let back-to-back turns synthesize on one socket while audio still plays in order.

NEW DETECTION Add `vad_force_turn_endpoint` to `ZoomAISTTService`: set it to `True` to disable Scribe's own VAD and commit turns on `VADUserStoppedSpeakingFrame`, letting Pipecat's VAD and turn strategies (including Smart Turn) control turn boundaries.

Control OpenAI Realtime long-session token truncation with the new `truncation` field on `SessionProperties`; set `TruncationRetentionRatio` to batch truncations and keep more input in the prompt cache, or `"disabled"` to return an error instead.

Format the diarization speaker into transcript text with the new `speaker_format` setting available on every STT service that supports speaker diarization.

Add Genesys pause/resume control with new GenesysAudioHookSerializer.create_pause_request() and create_resume_request() methods; sessions opened with `start_paused=True` now resume correctly once Genesys confirms.

+2 more

NEW INTEGRATION Sarvam HTTP STT integration

WAS THIS USEFUL? 👍 👎

◆ Model Training & Fine-Tuning

Google MaxText

Feedback ⓘ maxtext-v0.2.6 NOTES ↗ RANK

MaxText provides JAX implementations and distributed-training workflows for language models on TPUs and GPUs.

> MaxText's v0.2.6 release adds support for five new model architectures — DeepSeek-V4-Pro, GraniteMoE, GLM-5.3-Flash, Hy4-preview and Ming — plus Halo Sliding-Window Context Parallelism and a set of new MoE, Multi-Token Prediction and Splash Attention tuning configs.

└──▷ WHAT SHIPPED · 10 FEATURES most completely described first ? what's the number?

01 **GLM-5.3-Flash architecture support** NEW 85

MaxText adds the GLM-5.3-Flash architecture (`glm5.3-flash`) with configuration at `src/maxtext/m3/configs/glm5_3_flash_config.py` , a `dsa_attention_implementation="dense_mask"` dense-mask DSA attention mode, and training via `glm5_train.py` with float32 masked cross-entropy and AdamW parameter freezing.

— Exact config path, flag and training script given maxtext-v0.2.6

02 **DeepSeek-V4-Pro model support** NEW 80

MaxText adds a DeepSeek-V4-Pro (1.6T parameters, 384 routed experts) model configuration (`deepseek4-1600b.yml`) with HuggingFace checkpoint conversion and end-to-end test pipelines.

— Names config file, size and expert count, but no mechanism detail maxtext-v0.2.6

03 **Halo Sliding-Window Context Parallelism** NEW 80

Enables Halo Sliding-Window Context Parallelism for hybrid sliding-window models (Gemma 2/3/4, GPT-OSS, OLMo 3) via `local_context_parallel_strategy="halo"` on `AttentionOp` .

— Exact config flag and target models named maxtext-v0.2.6

04 **Splash Attention tuning flags** NEW 75

Adds `sa_ignore_segment_ids` to skip uniform segment mask overhead in Splash Attention for continuous stream-chunked datasets, and the experimental `sa_q_prescaled_log2e` and `sa_dkv_reuse_scaled_q` options to tune Splash Attention FP8 Q-scaling (both off by default).

— All three flags named with defaults and purpose maxtext-v0.2.6

05 **GraniteMoE architecture support in M3** NEW 70

MaxText adds GraniteMoE (`granite-3.0-1b-a400m`) architecture support in M3, including Flax NNX decoder layers, checkpoint parameter mappings, and the `model-bringup-m3` agent skill.

— Names components and skill but no usage mechanism `maxtext-v0.2.6`

06 **Hy4-preview architecture support** NEW 70

MaxText adds the Hy4-preview model architecture (`hy4-preview.yml`) combining Gated MLA, DeepSeek Sparse Attention indexers, Identity Hyper-Connections, and routed MoE feed-forward blocks, with HuggingFace checkpoint conversion.

— Names components but no behavioural detail beyond architecture list `maxtext-v0.2.6`

07 **DeepSeek-V4 router auxiliary loss config** NEW 70

Enable sequence-wise auxiliary load balancing loss for DeepSeek-V4 routers with the `moe_use_megatron_seq_aux_loss: true` config key.

— Exact config key and value given, minimal behavioural detail `maxtext-v0.2.6`

THINNER COVERAGE BELOW

08 **Ming hybrid KDA/MLA decoder architecture** NEW 55

MaxText adds the Ming hybrid KDA/MLA decoder architecture with scanned execution and the `ming-3.0-audio-tiny.yml` model configuration.

— Names config file but mechanism is only briefly described `maxtext-v0.2.6`

09 **Multi-Token Prediction decoder normalization config** NEW 55

Apply decoder normalization before the Multi-Token Prediction block by setting `mtp_norm_main_hidden_state` in the training config.

— Config key named, limited explanation of effect `maxtext-v0.2.6`

10 **Additional model architectures added** NEW 15

MaxText's summary also lists support for Qwen3.8-Flash-Next and WeaverOmni architectures, though no further configuration or implementation detail is given.

— Names only mentioned, no config or mechanism detail `maxtext-v0.2.6`

WAS THIS USEFUL? 👍 👎

fizgig

Feedback ⓘ v9.6.4 NOTES ↗ RANK

Fizgig — LoRA & Fine-tune Studio for Klein 9B, Krea 2, MiniMax H3, Qwen Image 2.1, Anima & Z-Image Turbo and Ace-Step XL: train, fine-tune, profile, repair and extract LoRAs & LoKR for Image, Video & Music Models

> Fizgig adds full training support for MiniMax H3 Image — including a dedicated training adapter, Clean-end share/Likeness tuning, Element boost, and 2048px previews — alongside out-of-memory auto-recovery, queue resilience, updated ROCm/torch defaults, and a new NVIDIA Sysmem Fallback startup check.

└──▷ **WHAT SHIPPED · 8 FEATURES** most completely described first ? what's the number?

01 **MiniMax H3 Image preset training tuning** IMPROVED 85

MiniMax H3 Image presets now train at 0.5 MP and start Automagic v3 at a learning rate of `2e-4`, skipping the initial climb phase; a Clean-end share setting in Training Parameters controls what fraction of training targets the final denoising step where skin, eyes and fine detail form, with presets defaulting to 40% versus the model's own default of approximately 17%, and this was superseded by a Likeness optimization that instead shifts training weight toward the final steps, on by default in the MiniMax H3 Image Standard preset and replacing the Clean-end share box, with the option to disable it on the Training tab under Training Parameters to restore H3 Image's even share.

— Names exact setting, defaults and disable location across both releases Gives exact numbers but no mechanism or UI path

v9.6.4 v9.6.0 v9.6.1

02 **ROCm and torch defaults for AMD installs** BREAKING 85

AMD installs now default to torch 2.14.0 for ROCm 10.1 from AMD's stable index; use `ROCM_CHANNEL=nightly` on Linux or `install_fizgig_rocm.bat --experimental` on Windows to keep nightly ROCm builds instead.

Keep ROCm nightly builds on a Linux AMD system instead of adopting the new stable default.

```
$ ROCM_CHANNEL=nightly ./install_fizgig.sh
```

Keep ROCm nightly builds on a Windows AMD system instead of adopting the new stable default.

```
$ install_fizgig_rocm.bat --experimental
```

— Exact env var, flag and versions make this directly runnable v9.6.0

03 **MiniMax H3 Image training support** NEW 80

Adds MiniMax H3 Image as a training target supporting LoRAs, LoKRs, edit LoRAs from before/after pairs, and slider LoRAs from photo pairs, with masked training, Multi Concept, DOP, Fizgig LoRA Isolation, batch size, and Compile Blocks; a training adapter is now on by default for MiniMax H3 Image, giving photographic skin and clearer likeness on the same dataset, downloading with the model, staying frozen during training, and not being saved into the output LoRA; and Element boost (from MOSS-Image) lets you weight face, hair, or another selected part more heavily early in a training run so likeness arrives sooner, off in every H3 Image preset by default and enabled by ticking On in Training Parameters.

— Lists supported outputs and controls but no exact command or menu path Explains mechanism and default state but gives no toggle or setting Names the toggle and its location in Training Parameters

v9.6.0 v9.6.1

04 **NVIDIA Sysmem Fallback startup diagnostic** NEW 70

Fizgig now shows startup steps on Windows with an NVIDIA card to set `CUDA - Sysmem Fallback Policy` to Prefer No Sysmem Fallback in NVIDIA Control Panel's Global Settings, with a Don't show this again option to dismiss it once configured.

— Gives exact setting name and location in NVIDIA Control Panel v9.6.3

05 **Out-of-memory auto-recovery for training runs** NEW 60

Out-of-memory recovery now automatically retries a failed web-UI run up to three times, stepping down through more block swap, then lower precision, continuing from the newest save when one exists.

— Explains retry mechanism but offers no manual control v9.6.0

THINNER COVERAGE BELOW

06 **2048-pixel preview resolution option** NEW 50

Studio and the Samples tab now offer 2048 as a Width and Height choice for every model, with MiniMax H3 Image defaulting to a 2048×2048 preview size.

— Names exact UI location and default resolution v9.6.1

07 **Preset reset on model switch** IMPROVED 50

Switching models on the Training tab now loads that model's first preset, clearing settings that do not apply to the new model, while keeping folders and the Samples tab.

— Names exact UI tabs affected by the behaviour change v9.6.0

08 **Queue resilience on failed runs** IMPROVED 35

A failed run no longer stalls the queue; Fizgig names the failed run in the Queue box and starts the next one automatically.

— Describes behaviour change with no configuration surface v9.6.0

WAS THIS USEFUL? 👍 👎

Google DeepMind Optax

Feedback ⓘ v0.2.9 NOTES ↗ RANK

Optax provides composable gradient transformations and optimization algorithms for JAX.

> Optax 0.2.9 adds weight decay support to several optimizers, expands the public API, and improves microbatching and numerical safety checks.

└──▷ **WHAT SHIPPED · 4 FEATURES** most completely described first ? what's the number?

01 Weight decay support in AdaBelief, DoWG-AdamW and Prodigy

NEW 80

The `adabelief` optimizer gains an optional `weight_decay` parameter, while `optax.contrib.dadapt_adamw` and `prodigy` gain a `weight_decay_mask` parameter that lets you restrict weight decay to a subset of parameters (e.g. excluding biases).

Apply weight decay alongside the AdaBelief adaptive moment estimates, previously unavailable in this optimizer.

python

```
optimizer = optax.adabelief(learning_rate=1e-3, weight_decay=1e-4)
```

Restrict weight decay to non-bias parameters in a DoWG-AdamW or Prodigy run using a mask.

python

```
optimizer = optax.contrib.dadapt_adamw(
    learning_rate=1e-3,
    weight_decay=0.01,
    weight_decay_mask={'w': True, 'b': False},
)
```

— Named optimizers and params with runnable code examples v0.2.9

THINNER COVERAGE BELOW

02 Trace-time error on eps underflow IMPROVED 45

Optax now raises a `ValueError` at trace time when `eps` underflows in the optimizer state dtype, catching a precision issue earlier than before.

— Names the check but no example of trigger v0.2.9

03 Size-0 batch support in microbatching IMPROVED 35

Microbatching now supports size-0 batches, enabling use cases where batch size is random and possibly empty.

— Describes behaviour but gives no usage example v0.2.9

04 **ArrayTree added to public API** NEW 25

`ArrayTree` is now exposed as part of Optax's public API.

— No explanation of purpose or usage shown v0.2.9

WAS THIS USEFUL? 👍 👎

AGENT

◆ Web Search & Scraping for Agents

Perplexity API

Feedback ⓘ updated · Nov 12, 2026 NOTES ↗

RANK

Perplexity API provides language models with real-time web search and retrieval for building applications that research and answer questions.

> Perplexity shipped automatic per-step output-token budgeting for the Agent API and introduced versioned effort levels that lock in specific token budgets.

└──▷ WHAT SHIPPED · 2 FEATURES most completely described first ? what's the number?

01 **Automatic per-step output-token limits in Agent API** IMPROVED

75

Developers can now omit `max_output_tokens` on any model, including Anthropic models, and each generation step automatically receives up to 128,000 output tokens, capped by the model's supported limit. Earlier steps no longer consume the next step's token budget.

— Names exact token cap and config key but no migration steps snapshot-20261112

02 **Versioned effort levels for Agent API** NEW

71

New effort-level versions `2026-11-11.0` now apply when calling an effort level by name (e.g. `effort="fast"`) for the levels `fast`, `low`, `medium`, `high`, `xhigh`, and `ultra`. Earlier pinned versions retain their previously configured budgets.

— Names version string and all effort levels, no runnable example snapshot-20261112

WAS THIS USEFUL? 👍 👎

◆ MCP Servers & Tooling

api-mcp-server

Feedback ⓘ v3.40.0 NOTES ↗ RANK

MCP server that exposes Hostinger's platform API for hosting deployment, CDN control, domain management, and WordPress operations.

> api-mcp-server added a set of PocketBase backend management operations and bundled agent skills for headless frontend and validation workflows.

└──▷ **WHAT SHIPPED · 2 FEATURES** most completely described first ? what's the number?

01 **PocketBase backend management operations** NEW 80

Five new operations for managing PocketBase backends on hosting accounts: `hosting_pocketbase_backend` to get a backend, `hosting_pocketbase_create-backend` to create one, `hosting_pocketbase_delete-backend` to delete one, `hosting_pocketbase_list-backends` to list all account backends, and `hosting_pocketbase_set-superuser` to configure a superuser.

— Names all five operations exactly, but no mechanism detail v3.39.0

02 **Bundled agent skills for headless workflows** NEW 60

Adds bundled agent skills for headless frontend and validation steps, exposed via `skills/headless/VERSION`, `skills/headless/steps/06-frontend/SKILL.md`, and `skills/headless/steps/07-validate/SKILL.md`.

— Names file paths but not what the skills actually do v3.40.0

WAS THIS USEFUL? 👍 👎

godot-mcp

Feedback ⓘ v0.39.0 NOTES ↗ RANK

Godot-MCP — Model Context Protocol (MCP) integration for the Godot Engine. dev.

> Godot-MCP's v0.39.0 release adds a full set of headless CLI subcommands for CI-style Godot operations, plus new `tools`, `rql`, and `setup-skills` commands and a shared exit-code system for agent error handling.

└──▷ **WHAT SHIPPED · 5 FEATURES** most completely described first ? what's the number?

01 **Headless CLI subcommands for check, import, test, export, run-script**

NEW 73

Run headless Godot operations from the CLI without opening the editor using the new `check`, `import`, `test`, `export`, and `run-script` subcommands, all sharing a common exit-code scheme and a `--format` flag to control output format.

— Names all five subcommands and the `--format` flag but gives no example invocation

v0.39.0

02 **Shared exit-code system for agent error classification**

IMPROVED 71

Tool call failures are now classified using a shared cli-core exit-code table so agents can programmatically distinguish error categories; a dropped mid-call connection now exits with code `7` (`UNREACHABLE`) via the shared `networkFailureReason` table instead of a generic error.

— Names exact exit code and tables but no usage walkthrough v0.39.0

03 **tools command for MCP tool discovery** NEW 65

The new `tools` command lets agents discover available MCP tools from the terminal, with agent-discovery improvements validated against DeepSeek CLI trials.

List every MCP tool available to an agent so you can verify what capabilities are exposed before wiring up an AI workflow.

```
$ godot-cli tools ./MyGodotProject
```

— Named command with a runnable example invocation v0.39.0

04 **setup-skills for offline cold-start agent skill** NEW 65

The `setup-skills` command writes a cold-start godot-cli skill for offline environments, and also renames the addon guide to `godot-cli-install`.

Bootstrap a cold-start agent skill for offline environments so the agent knows how to operate godot-cli before the editor is reachable.

```
$ godot-cli setup-skills ./MyGodotProject
```

— Named command plus renamed guide, backed by runnable example v0.39.0

THINNER COVERAGE BELOW

05 rql command for token-efficient agent queries NEW

38

The new `rql` command reduces token usage per agent task, with short help text including recipes and plain run-tool answers.

— Names the command but no example or further mechanism v0.39.0

WAS THIS USEFUL? 👍 👎

claudexor

Feedback ⓘ v6.0.0 NOTES ↗ RANK

Claudexor is a multi-harness control plane for Claude Code, Codex, Cursor, and OpenCode with quota-aware rotation across multiple subscriptions.

> Claudexor 6.0.0 adds a full harness-program management CLI, async blocking questions, and per-lane effort controls on continuation, while bumping the engine store schema and control protocol and tightening installation flags, distribution, and runtime version requirements.

└──▷ **WHAT SHIPPED · 10 FEATURES** most completely described first ? what's the number?

01 **Harness vendor-program management CLI** NEW 95

`claudexor harness inspect --latest --json` inspects all seven local harness programs, reporting installed and published versions, action command previews, release-note links, and `updateAvailable` status of `yes`, `no`, or `cannot_tell`. `claudexor harness install <id> --yes` installs a missing harness program and `claudexor harness update <id> --yes` updates one, with `--vendor-version <exact>` available when the adapter offers an exact-version command; `claudexor harness install --dry-run --json` previews a maintenance operation and returns its disclosure without executing it.

Check which of your harness programs have updates available before deciding whether to update them.

```
$ claudexor harness inspect --latest --json
```

Install a missing harness program (for example, the Codex harness) in one unattended step.

```
$ claudexor harness install codex --yes
```

Preview what a harness install operation would do without executing it, to review the maintenance disclosure first.

```
$ claudexor harness install codex --dry-run --json
```

— Full command surface and flags named with runnable examples v6.0.0

02 **Engine store schema and protocol version bump** BREAKING

80

The control protocol major version is now 3 and the engine store schema is now 8, adding each question's `delivery` field (`blocking` for earlier rows) and an index of pending questions; opening an existing store migrates it from schema 7 automatically, but the migration is forward-only and an older engine refuses a data root served by this release — replacing its executable is not a rollback.

— Full mechanism and consequence named, no manual recovery step given v6.0.0

03 **Async blocking questions** NEW

78

Questions now declare `delivery: blocking | async` ; async questions (`asyncQuestionsOnly`) let the agent keep working, set `waitingOnUser` , and wake `claudexor_run_wait` while still accepting answers through ordinary routes.

— Names fields and wake signal but gives no CLI example v6.0.0

04 **Harness installation flag and disclosure changes** BREAKING

75

Harness installation and maintenance commands now refuse the `--target` and `--release-tested` flags, the `baseline` and `previous` maintenance targets, and `--vendor-version latest` (omit the flag to get the vendor's current update); old verification classes and pinned-version disclosure fields are no longer present in inventory and receipts.

— Names every refused flag and removed field but no migration guidance v6.0.0

05 **Per-lane effort controls on continuation** NEW

73

Passing `effort` or a harness-keyed `efforts` map to `claudexor_run_continue` sets per-lane effort on continuation; a lane uses its own map entry first, then falls back to the scalar `effort` , then to Settings.

— Names the field and fallback order, no example invocation v6.0.0

06 **Model catalog provenance and model-name reporting** NEW

65

Program-derived model catalogs now carry a `sourceProgram` path and version, failed model operations expose `problem.context.requestedModel` , and `route.model` names only a provider-reported model.

— Names the exact fields but no retrieval command shown v6.0.0

THINNER COVERAGE BELOW

07 **Antigravity read-only session hardening** IMPROVED

57

Antigravity read-only runs now require `agy` 1.4.10 or newer and use session settings that deny commands and file writes while preserving the profile's allow/deny/ask rules.

— Names version requirement and behavior, no setup steps v6.0.0

08 **Distribution and runtime version requirements** BREAKING

55

Hosts no longer receive an npm installer, and local and SSH runtime manifests now require Claudexor.app 6.0.0 or newer; an older app reports that an update is required.

— States the requirement but no upgrade path detail v6.0.0

09 **OAuth org-not-allowed error mapping** IMPROVED

47

Claude's `oauth_not_allowed_for_organization` code now maps to `oauth_org_not_allowed`, preserving the vendor cause and setting the account aside for its existing refusal period.

— Names the exact codes but no action a user takes v6.0.0

10 **Receipts and idempotency retention window** IMPROVED

38

Model-operation receipts and idempotency bindings now persist for 30 days after termination, with no cap on Agent history count.

— States the new window but no config or command to adjust it v6.0.0

└──▷ **BREAKING ON UPGRADE**

! Installation `--target` and `--release-tested` flags are refused; use the supported spellings instead.

! Maintenance `baseline` and `previous` targets are refused.

! `--vendor-version latest` is refused; omit the flag to get the vendor's current update.

! Old verification classes and pinned-version disclosure fields are absent from inventory and receipts.

! Hosts no longer receive an npm installer.

! Local and SSH runtime manifests require Claudexor.app 6.0.0; an older app reports that an update is required.

! The control protocol major is now 3 and the engine store schema is now 8; an older engine refuses a data root served by this release, and replacing its executable is not a rollback.

! Engine store migration from schema 7 to 8 is forward-only and adds `delivery` (`blocking` for earlier rows) and an index of pending questions.

WAS THIS USEFUL?  

pullmd

Feedback ⓘ v3.19.0 NOTES ↗ RANK

Pullmd is a self-hosted URL- and file-to-Markdown service for humans and AI agents - web pages, documents, images, audio, YouTube. PWA + REST + MCP + Claude Code skill, Reddit-aware, refreshable share links.

> Pullmd added an async file-conversion API so long-running jobs like audio transcription and PDF OCR can survive mobile app backgrounding instead of being cancelled.

└──▷ **WHAT SHIPPED · 1 FEATURE** most completely described first ? what's the number?

01 **Async file conversion API** NEW 95

Passing `async=1` to `POST /api/file` now returns `202` with a job id instead of blocking, covering long jobs like audio transcription and PDF OCR that a mobile OS would otherwise cancel. Poll job state or retrieve the finished Markdown with `GET /api/file/jobs/{id}`, and clean up with `DELETE /api/file/jobs/{id}`. When the queue is full, `POST /api/file?async=1` falls back to synchronous conversion and returns `200`, so clients must handle both `202` and `200`. Behaviour is tunable via `PULLMD_FILE_JOB_TTL_MINUTES` (how long finished results wait in memory, up to 60 minutes; `0` disables async jobs entirely) and `PULLMD_FILE_JOB_MAX_MB` (memory cap per job, default 64 MB).

Submit a large PDF for OCR from a mobile client and immediately get a job id you can poll later, without keeping the connection open.

```
$ curl -X POST 'https://your-pullmd-instance/api/file?async=1' \
  -F 'file=@document.pdf' \
  -H 'Authorization: Bearer <token>'
```

Poll for the finished Markdown once you return to the app, using the job id from the 202 response.

```
$ curl 'https://your-pullmd-instance/api/file/jobs/<id>' \
  -H 'Authorization: Bearer <token>'
```

Tune how long finished job results stay in memory and cap the per-job memory footprint for a self-hosted instance.

```ini
PULLMD_FILE_JOB_TTL_MINUTES=30
PULLMD_FILE_JOB_MAX_MB=64
```

— Full endpoint set, status codes, and env vars with defaults given v3.19.0

WAS THIS USEFUL? 👍 👎

task-orchestrator

Feedback ⓘ v4.0.2 NOTES ↗ RANK

task-orchestrator is an MCP server that enforces workflow discipline for AI agents with persistent work tracking.

> task-orchestrator's two releases this window focus on HTTP MCP session lifecycle hardening and a large cleanup of token/cost budget tracking from the runs API, alongside a small additive enhancement to run frontier reporting.

└──▷ **WHAT SHIPPED · 3 FEATURES** most completely described first ? what's the number?

01 Removal of token/cost budget tracking across runs API

96

BREAKING

task-orchestrator strips token and cost budget tracking from the MCP and REST surface: the `advance` `heartbeat` and `usage` operations are removed (request fails `unknown_parameter`); `runs_write` `open` / `update` fields `budget.tokens` and `budget.costUsd` are removed (`unknown_parameter` on `/budget/tokens` or `/budget/costUsd`); `runs_write` `answer` field `amount` is removed (`unknown_parameter` on `/amount`); run view members `usage.reportedTokens`, `usage.reportedCostUsd`, `budget.tokens`, and `budget.costUsd` are removed from `runs_read` responses; the `usage` member of a `runs_read` `seats` row is removed; the `paused: {decisionId}` member of write results is removed; the `budget` section of the `report` run rollup is removed; the `run.opened` event's `budget` payload no longer carries `tokens` or `costMicroUsd`, carrying only attempt and seat caps. The plugin flags `--run-budget-tokens` and `--run-budget-usd` are removed from `ralph-loop` (passing them now exits 64), and `--phase heartbeat` and `--usage` are removed from `run-ops`. A config document that still sets `runs.budgets.tokens` or `runs.budgets.cost_usd` loads but each key draws one warning and no ceiling is applied.

— Enumerates every removed field, endpoint, flag, and config key v4.0.1

02 HTTP MCP session idle timeout control and SDK 0.16.0 upgrade

88

BREAKING

task-orchestrator adds `MCP_SESSION_IDLE_TIMEOUT_MINUTES` (default `120` minutes) to cap how long an HTTP MCP session may sit idle before the SDK closes it; a blank, zero, negative, or non-integer value fails startup and names the variable. The bundled MCP Kotlin SDK is upgraded to 0.16.0: the default idle session lifetime for HTTP clients drops from 24 hours to 2 hours (clients relying on long idle sessions now get a `404` and must re-initialize), a `GET` to `/mcp` without `Mcp-Session-Id` returns `400` instead of succeeding, opening a new session while 10,000 are already active returns

`503`, and a `POST` without a session id that isn't `initialize` no longer leaves a dangling session behind.

Tighten idle session lifetime to 30 minutes in environments where stale HTTP MCP sessions should be reclaimed faster than the 120-minute default.

```env
MCP_SESSION_IDLE_TIMEOUT_MINUTES=30
```

— Names env var, default, and exact HTTP status codes/behaviors v4.0.2

---

03 **Cancelled counts and item titles in runs_read frontier**

IMPROVED 70

The `runs_read` `frontier` operation now reports a `cancelled` count of terminal members with a cancel status label, and each `inFlight` entry gains the item's `title`. Both fields are additive on MCP and on `POST /api/v1/ops/runs_read`.

Check cancelled member counts and in-flight item titles for a run on the REST API.

```
$ curl -X POST https://<host>/api/v1/ops/runs_read \
  -H 'Content-Type: application/json' \
  -d '{"operation": "frontier", "runId": "<run-id>"}'
```

— Names fields and endpoint but mechanism is brief v4.0.1

└──▷ **BREAKING ON UPGRADE**

! A `GET` to `/mcp` without `Mcp-Session-Id` now returns `400` instead of succeeding (MCP Kotlin SDK 0.16.0 behavior).

! The default idle session lifetime for HTTP clients drops from 24 hours to 2 hours; clients that relied on long-lived idle sessions will now receive a `404` and must re-initialize.

! The `advance` `heartbeat` and `usage` operations are removed from the MCP and REST surface; a request that sends a removed field fails `unknown_parameter` on that field.

! `runs_write` `open` / `update` fields `budget.tokens` and `budget.costUsd` are removed; sending them returns `unknown_parameter` on `/budget/tokens` or `/budget/costUsd`.

! `runs_write` `answer` field `amount` is removed; sending it returns `unknown_parameter` on `/amount`.

! The run view members `usage.reportedTokens`, `usage.reportedCostUsd`, `budget.tokens`, and `budget.costUsd` are removed from `runs_read` responses.

! The `usage` member of a `runs_read` `seats` row is removed.

! The `paused: {decisionId}` member of write results is removed.

! The `budget` section of the `report` run rollup is removed.

! The `run.opened` event's `budget` payload no longer carries `tokens` or `costMicroUsd` — it now carries attempt and seat caps only.

! The plugin flags `--run-budget-tokens` and `--run-budget-usd` are removed from `ralph-loop`; passing them now exits 64.

! The plugin flags `--phase heartbeat` and `--usage` are removed from `run-ops`.

! A config document that sets `runs.budgets.tokens` or `runs.budgets.cost_usd` still loads but each key draws one warning and no ceiling is applied.

WAS THIS USEFUL?  

◆ Agent Sandboxes

containarium

Feedback ⓘ v0.136.0 NOTES ↗ RANK

Containarium is an open-source agent runtime — SSH-native isolation, eBPF egress policy, Kubernetes + LXC backends, GPU passthrough, MCP-native CLI

> Containarium adds a `--workdir` option for code execution and now surfaces refused audit anchors across health and monitoring endpoints.

└──▷ **WHAT SHIPPED · 2 FEATURES** most completely described first ? what's the number?

01 **Working directory flag for code run** NEW 75

Set a working directory for code execution using `--workdir` on the `code run` command or the `workdir` parameter in the MCP `code_run` tool, so relative paths and project-local configs resolve correctly inside the container.

Run code in a specific working directory inside a container, so relative paths and project-local configs resolve correctly.

```
$ containarium code run --workdir /workspace/myproject script.py
```

— Names exact flag, MCP param, and shows runnable command v0.136.0

THINNER COVERAGE BELOW

02 **Refused audit anchor reporting** NEW 50

containarium now reports a refused audit anchor in `/health`, `doctor`, and metrics endpoints.

— Names surfaces but not what anchor represents or why refused v0.136.0

WAS THIS USEFUL? 👍 👎

◆ AI Agent Frameworks

cotal

Feedback ⓘ v0.161.0 NOTES ↗ RANK

Cotal is the open standard for agent coordination

> Cotal's biggest shipment this window is a live account resolver that lets a running broker add and retire space accounts without a reload, alongside multi-connector model listing via `cotal models` and a string of daemon/desktop improvements covering computer vitals, update management, account session controls, and status visibility.

└──▷ **WHAT SHIPPED · 11 FEATURES** most completely described first ? what's the number?

01 Live account resolver for broker space accounts NEW

98

Start a broker with `cotal up --account-resolver full` to add and retire space accounts on a live broker without reloading config, and roll back with `cotal up --account-resolver memory`. Under the hood, `serverConfig` accepts `accountResolver: { kind: "full", directory, limit, intervalMs }` to render the nats-server directory resolver, `pushAccountClaim` sends a signed account claim over a system-account connection and waits for every expected server to accept it, and `retireAccountClaim` re-signs an account so it serves nobody.

Start a broker that can accept pushed account claims at runtime — no config re-render or reload needed when adding or retiring space accounts.

```
$ cotal up --account-resolver full
```

Roll back the full resolver choice on a stopped root so the broker returns to in-memory account resolution.

```
$ cotal up --account-resolver memory
```

Switch a running broker from the default MEMORY resolver to the directory resolver so new space accounts can be pushed and retired without a broker reload.

```yaml
accountResolver:
  kind: "full"
  directory: "/var/cotal/accounts"
  limit: 1000
  intervalMs: 5000
```

— Full config shape, flags and runnable commands all named `v0.159.0` `v0.160.0`

02 **Model listing via `cotal models` across connectors** NEW

95

List available models with `cotal models`: for Claude Code it queries your local `claude` binary via its stream-json control protocol without sending a prompt; `cotal models --harness opencode` runs `opencode api --standalone GET /api/model` and returns each model as a `provider/model` ID with its variant names; for pi devices it opens a short-lived `pi --mode rpc --no-session --offline` session that sends one `get_available_models` command and returns each model as a `provider/id`.

Enumerate which Claude Code models your account can use before assigning one to an agent session.

```
$ cotal models
```

Check which models your local OpenCode 2.x installation exposes, returned as `provider/model` IDs, before configuring a harness.

```
$ cotal models --harness opencode
```

— Three runnable commands with exact flags and protocols named `v0.160.1`

03 **New daemon status and heartbeat fields** NEW

90

Adds new daemon status reporting: the `machineService` field in `cotal daemon status --json` output (which no longer blocks a space from reading as connected), computer vitals — CPU, memory, home-volume storage and uptime — reported through the daemon's account heartbeat and local status, and visibility into the agents running on this computer in `daemon status` output and the desktop app by reading bounded public seat facts from each hosted Manager over its parent channel.

Inspect machine-service approval status separately from overall space connectivity in scripts or dashboards.

```
$ cotal daemon status --json | jq '.machineService'
```

— Named field, command and runnable jq example given States metrics and surface but no command to inspect them Names the command and data source, lacks a worked example

v0.160.6 v0.161.0 v0.160.8

04 **Per-message read-state tracking in terminal app** NEW

85

Read and record per-message read state in the terminal app using readMarks() and markRead() from `@cotal-ai/workspace/chat`, persisted in a private `read.json` beside a space's chat sessions and capped at the newest 1000 entries; `cotal space forget` removes it alongside the sessions.

— Named functions, module and file, no runnable example given v0.160.0

05 **Typed rate-limit error for token reads** NEW

65

Token reads that hit provider rate limits now expose the HTTP 429 status and an optional `retryAfterSeconds` field via the typed `IdpRateLimitedError` error, without adding artificial retries.

— Named error type and field, no usage example shown v0.161.0

06 **Update preference and auto-update management** IMPROVED

65

Manage update preferences and view available desktop releases in Cotal Settings, with the daemon exposing preference and cached-check requests shared with the update CLI; unattended updates are now allowed on computers that haven't run setup, by creating missing protected config folders before the update snapshot. Updates can also be checked for on every platform, and a Linux machine daemon can be kept current through the guarded updater, with independent controls for automatic installation (Linux only in this release) and notices — manual updates remain available on every platform.

— Names UI path and platform scope but no flags or commands v0.159.1 v0.160.8

THINNER COVERAGE BELOW

07 **Credential cleanup on daemon disconnect or uninstall** IMPROVED

55

Disconnecting or uninstalling the daemon now removes its local service credentials and identity records; removal failures exit non-zero and retain the installation receipt for retry.

— Describes exact failure/retry behavior, no command example v0.161.0

08 **Account sign-out and sign-in session controls** IMPROVED

50

Add confirmed sign-out and disconnect controls to desktop Settings, sharing the CLI's provider-session revocation and local sign-in cleanup while leaving existing computer connections running; daemon sign-in now names the replaced account in human and plain output when switching accounts at the same provider.

— Describes UI and CLI behavior without exact command v0.160.8

09 **Reconnect warning for stalled channel delivery** IMPROVED

50

Reconnect now warns when ambient channel delivery still requires a session restart, distinguishing transport recovery from exhausted channel drop tracking.

— Explains the mechanism distinction, no exact command shown v0.160.8

10 **Directory browsing for agent launch location** NEW 40

Browse directories on your computer when choosing where to launch an agent, served through the same authorized control path as working-directory validation.

— Describes the feature without a command or flag name v0.160.8

11 **Computer naming before and after connecting** NEW 40

Name a computer before connecting to Cotal and change its saved name later from Settings or the command line.

— No exact command or flag named for the CLI path v0.160.2

WAS THIS USEFUL?  

assistant-ui

Feedback ⓘ `@assistant-ui/react@0.15.36` NOTES ↗

RANK

Assistant-ui provides React components and state management for chat interfaces with streaming, tool calls, and attachments.

> assistant-ui shipped a new focus-guard hook for portaled composers and raised the peer dependency floor for its Lexical integration.

└──▷ **WHAT SHIPPED · 2 FEATURES** most completely described first ? what's the number?

01 **Composer run-start focus guard hook** NEW 69

The new `unstable_useComposerRunStartFocusGuard` hook from `@assistant-ui/react` lets a composer that is portaled outside `ThreadPrimitive.Root` receive focus when a thread run starts and focus is already inside its thread, matching the existing behavior of `ComposerPrimitive.Input`.

— Names exact hook and behavior but no usage example `@assistant-ui/react@0.15.36`

THINNER COVERAGE BELOW

02 **Lexical package peer dependency bump** BREAKING 50

`@assistant-ui/react-lexical` now requires `@assistant-ui/react` peer version `^0.15.36` or later, since that is the first release exporting `unstable_useComposerRunStartFocusGuard`; older `@assistant-ui/react` versions are no longer accepted.

— States new peer floor but no migration steps `@assistant-ui/react@0.15.36`

└──▷ **BREAKING ON UPGRADE**

! `@assistant-ui/react-lexical` raises its `@assistant-ui/react` peer floor to `^0.15.36`; older versions of `@assistant-ui/react` are no longer accepted by this package.

WAS THIS USEFUL? 👍 👎

PydanticAI

Feedback ⓘ v2.68.0 NOTES ↗ RANK

PydanticAI is a framework that builds type-safe Python agents with dependency injection, model integrations, tools, and structured outputs.

> PydanticAI's v2.68.0 release adds a Gemini Decisions API backend, a ConversationStore for UI adapters, automatic model fallback on truncated output, mid-run user approvals via RequestInput, and durable Temporal/DBOS workflow support for sub-agent delegations, alongside several breaking renames in the harness and sub-agent event APIs.

└──▷ **WHAT SHIPPED · 13 FEATURES** most completely described first ? what's the number?

01 Automatic model fallback on truncated output NEW 85

Fall back to the next model in `ModelFallback` automatically when a response is cut off at `max_tokens`, using the new `OutputTokenLimitExceeded` exception.

Automatically try the next model when a response is truncated at max_tokens, rather than surfacing an error to the caller.

python

```python
from pydantic_ai import Agent
from pydantic_ai.models.fallback import ModelFallback

agent = Agent(
    ModelFallback(
        'openai:gpt-6.1-sol',
        'anthropic:claude-fable-5-1',
    )
)
# If gpt-6.1-sol hits max_tokens, OutputTokenLimitExceeded
triggers
# and ModelFallback automatically retries with claude-fable-5-1.
result = agent.run_sync('Summarize this very long document ...')
print(result.output)
```

— Runnable example with exact exception and config names v2.68.0

02 In-run user approvals with RequestInput NEW 85

Request approvals and answers during a run with `RequestInput`; deferred calls are now marked with `kind` and `answered_by` fields, and `AskUser` moves to the deferred pipeline.

Prompt the user for approval or an answer mid-run without leaving the agent loop, using RequestInput.

```python
from pydantic_ai import Agent
from pydantic_ai.request_input import RequestInput

agent = Agent(
    'anthropic:claude-fable-5-1',
    capabilities=[RequestInput()],
)
result = agent.run_sync('Deploy the staging environment.')
print(result.output)
```

— Runnable example and named fields/capability v2.68.0

---

03 **Durable Temporal and DBOS workflows for SubAgents** NEW

70

Add Temporal and DBOS durable workflow support for `SubAgents` delegations, including running delegations to durable delegates as Temporal and DBOS child workflows, and cancelling a Temporal durable run from a workflow signal using `CancellationToken`.

— Concrete mechanism but no exact command shown v2.68.0

---

04 **Harness field and event renaming** BREAKING

70

In `pydantic_ai_harness`, `num_results` is renamed to `max_results` and `max_duration_secs` is renamed to `max_duration_seconds`; `CapabilityEvent` fields are renamed to `*_count`, `*_name`, `duration_ms`, and `previous_item`; `CapabilityEvent` outlier event classes are renamed to the `<Subject><PastParticiple>Event` pattern, and `STEP_PERSISTENCE_EVENTS` is added. Old names remain as deprecated aliases.

— Enumerates every renamed field for migration v2.68.0

---

05 **ConversationStore for UI adapters** NEW

60

Save and load UI adapter conversations by passing a `ConversationStore` via the `conversation_store` parameter.

— Names the parameter but no runnable example v2.68.0

---

06 **DelegationTasks.discover() method** NEW

60

Add DelegationTasks.discover() to find available delegation tasks without blocking on a damaged saved task.

— Named method with clear purpose, no example v2.68.0

---

07 **Sub-agent task-control event renames** BREAKING 60

Sub-agent task-control events are renamed with an `Event` suffix and past-participle names, and now carry `agent_name` and `parent_task_id` fields; old names remain as deprecated aliases.

– Notes new fields and deprecation path for migration v2.68.0

THINNER COVERAGE BELOW

08 **Per-command working directory in Workspace.run** NEW 55

Set a per-command working directory in `Workspace.run` with the new `cwd` parameter.

– Names exact method and parameter v2.68.0

09 **Streamed cost accuracy for Google and Bedrock models** IMPROVED 50

Stream accurate costs for `GoogleModel` and `BedrockConverseModel` from the AI Gateway's streamed cost data.

– Names models but no mechanism detail v2.68.0

10 **Workspace backends must leave processes detached** BREAKING 50

Workspace backends are now required to leave detached processes running, enforced by `WorkspaceBackendSuite`.

– States requirement but no migration steps v2.68.0

11 **Durable PromptInjectionDefender classification** BREAKING 50

`PromptInjectionDefender` semantic classification now runs as durable operations, changing its execution model when used on Temporal.

– Notes behaviour change but no concrete migration guidance v2.68.0

12 **GoogleDecisionsModel backend** NEW 45

Add a `GoogleDecisionsModel` backend for the Gemini Decisions API.

– Names the backend but no usage detail v2.68.0

13 **RefineOutput harness capability** NEW 40

Add a `RefineOutput` harness capability for iterative output refinement.

— Bare description with no mechanism v2.68.0

└──▷ **BREAKING ON UPGRADE**

! Sub-agent task-control events are renamed with an `Event` suffix and past-participle names, and now carry `agent_name` and `parent_task_id` fields; old names remain as deprecated aliases.

! In `pydantic_ai_harness`, `num_results` is renamed to `max_results` and `max_duration_secs` is renamed to `max_duration_seconds`; old names remain as deprecated aliases.

! Harness `CapabilityEvent` fields are renamed to `*_count`, `*_name`, `duration_ms`, and `previous_item`; old names remain as deprecated aliases.

! Harness `CapabilityEvent` outlier event classes are renamed to the `<Subject><PastParticiple>Event` pattern; `STEP_PERSISTENCE_EVENTS` is added.

! Workspace backends are now required to leave detached processes running, enforced by `WorkspaceBackendSuite`.

! `PromptInjectionDefender` semantic classification now runs as durable operations, changing its execution model when used on Temporal.

WAS THIS USEFUL? 👍 👎

CopilotKit

Feedback ⓘ v1.87.0 NOTES ↗ RANK

CopilotKit connects AI agents to application interfaces with shared state, tools, and generative UI components.

> CopilotKit added support for rendering A2UI 1.0 operations alongside the existing v0.9 protocol, letting agents that emit native 1.0 surfaces render correctly without breaking prior integrations.

└──▷ **WHAT SHIPPED · 1 FEATURE** most completely described first ? what's the number?

01 **A2UI 1.0 rendering support** NEW 40

CopilotKit now renders A2UI 1.0 operations alongside the existing v0.9 protocol, so agents emitting native A2UI 1.0 surfaces (such as those from ADK) display correctly without breaking existing integrations.

— Names protocol versions and source but no API or config detail v1.87.0

WAS THIS USEFUL? 👍 👎

CrewAI

Feedback ⓘ 1.15.40 NOTES ↗ RANK

CrewAI is an open-source Python framework for building multi-agent workflows using role-based AI agent Crews and event-driven Flows.

> CrewAI's 1.15.40 release adds JSONL recognition to RAG data discovery, switches standalone agent scaffolding to JSON by default, enables row-level historical agent state lookups, and decouples the embedding stack from ChromaDB.

└──▷ **WHAT SHIPPED · 4 FEATURES** most completely described first ? what's the number?

01 **JSONL support in RAG data discovery** NEW 40

JSONL files are now recognized as a supported input format in RAG data type discovery, wiring JSONL into the existing data type detection logic.

– Names the format but not detection mechanism or limits 1.15.40

02 **ChromaDB removed from embedding function layer** BREAKING 40

ChromaDB has been removed from the embedding function layer, decoupling the embedding stack from that dependency.

– Names the dependency removed but not replacement or migration steps 1.15.40

03 **Historical agent state lookup by row ID** NEW 35

Added the ability to look up exact historical agent state by row ID via the backend record lifecycle layer.

– Describes capability but no API or command shown 1.15.40

04 **JSON scaffolding for standalone agents** IMPROVED 30

CrewAI now scaffolds standalone agents as JSON by default instead of the previous format.

– States the change but not prior format or migration path 1.15.40

WAS THIS USEFUL? 👍 👎

openexecutive

Feedback ⓘ v0.7.2 NOTES ↗

Openexecutive is an ai-powered virtual executive team — a single coherent executive persona backed by 8 specialist agents (FastAPI + Next.js).

└──▷ GET THIS VERSION ▾

```
$ git clone --branch v0.7.2
https://github.com/SenteLabsAI/OpenExecutive.git
# already have the repo? check out this version:
$ git checkout v0.7.2
```

NEW INTEGRATION Google Chat sender identity on People

WAS THIS USEFUL? 👍 👎

HeyGen HyperFrames

Feedback ⓘ v0.9.34 NOTES ↗ RANK

HyperFrames is an open-source HTML-to-video renderer that runs in AI-agent workflows.

> HyperFrames added opt-in strict variable validation to catch undeclared render variables before they cause silent failures.

└──▷ WHAT SHIPPED · 1 FEATURE most completely described first ? what's the number?

01 **Strict variable checking in Producer** NEW 58

A new strict variable checking option in the Producer fails a render when it receives a variable the page does not declare. It is off by default, so existing renders are unaffected.

— Names the exact setting and default but no enable syntax v0.9.34

WAS THIS USEFUL? 👍 👎

AutoGPT

Feedback ⓘ autogpt-platform-beta-v0.11.2 NOTES ↗

AutoGPT automates multi-step workflows with continuous AI agents that users can build and deploy.

AutoGPT Platform adds self-serve account deletion, an erase-account CLI, E2B sandbox concurrency limits, in-app feedback prompts, and PostHog usage-limit and plan-tier events.

└──▷ GET THIS VERSION ▾

```
$ git clone --branch autogpt-platform-beta-v0.11.2
https://github.com/Significant-Gravitas/AutoGPT.git
# already have the repo? check out this version:
$ git checkout autogpt-platform-beta-v0.11.2
```

Adds self-serve account deletion with an optional exit question.

Erases closed accounts after 14 days using the new `erase-account` CLI and job, keeping a ban record.

Limits how many E2B sandboxes run at once, per deployment and per account.

Sends a PostHog event whenever a usage limit refuses or pauses work.

Sets the plan tier on the PostHog person.

+1 more

NEW INTEGRATION Wix MCP catalog entry

NEW INTEGRATION HubSpot OAuth sign-in for HubSpot blocks

NEW INTEGRATION Microsoft / Microsoft 365 sign-in and onboarding

NEW SUPPORTED PLATFORM Meta Muse Ember model support

NEW SUPPORTED PLATFORM Claude Fable 5.2 model support

WAS THIS USEFUL? 👍 👎

LangChain

Feedback ⓘ `langchain-typesafe==0.1.0` NOTES ↗ RANK

LangChain is an open-source framework that orchestrates applications powered by language models.

> LangChain's 1.5.0 wave introduces decision models and a new type-safe classifier, adds automatic model routing middleware, and drops Python 3.10 support across the core, main and typesafe packages.

└──▷ **WHAT SHIPPED · 3 FEATURES** most completely described first ? what's the number?

01 **Python 3.10 support dropped across packages** BREAKING

80

`langchain` 1.5.0, `langchain-core` 1.8.0, and `langchain-typesafe` 0.1.0 all now require Python 3.11 or later, with minimum `langchain-core` 1.8.0 (`langchain` 1.5.0 for the `experimental` extra). Projects remaining on Python 3.10 will continue to resolve the prior releases instead: `langchain` 1.4.x, `langchain-core` 1.7.x, and `langchain-typesafe` 0.0.x.

– Exact version pins and migration fallback given `langchain-typesafe==0.1.0` `langchain==1.5.0` `langchain-core==1.8.0`

THINNER COVERAGE BELOW

02 **Decision models in langchain-core and TypeSafeClassifier** NEW

55

`langchain-core` 1.8.0 adds decision models for structured decision-making in agent workflows, and the new `langchain-typesafe` 0.1.0 package introduces `TypeSafeClassifier` as a decision model built for type-safe classification workflows.

– Names the class and package but no usage shown `langchain-core==1.8.0` `langchain-typesafe==0.1.0`

03 **Model router and auto mode middleware** NEW 55

`langchain` 1.5.0 adds `ModelRouterMiddleware` and an auto mode middleware that route requests across models automatically.

– Names the middleware class, no config example given `langchain==1.5.0`

└──▷ **BREAKING ON UPGRADE**

! `langchain-typesafe` 0.1.0 drops Python 3.10 support; the minimum supported version is now Python 3.11 and the minimum `langchain-core` version is now 1.8.0 (`langchain` 1.5.0

for the `experimental` extra). Users on Python 3.10 will continue to resolve `langchain-typesafe` 0.0.x.

! Python 3.10 is no longer supported; the minimum supported version is now Python 3.11, and the minimum `langchain-core` version is now 1.8.0. Users on Python 3.10 will continue to resolve `langchain` 1.4.x.

! Python 3.10 support is removed; the minimum supported Python version is now 3.11. Projects on Python 3.10 will be pinned to `langchain-core` 1.7.x and will not receive 1.8.x updates.

WAS THIS USEFUL? 👍 👎

CopilotKit OpenBot

Feedback ⓘ v0.3.4 NOTES ↗ RANK

OpenBot runs self-hosted AI coworkers that each get their own browser, files, and gated tools, with every action decided and recorded.

> OpenBot's v0.3.4 release reworks Automatic Learning with per-person skill approval and sharing controls, adds a model-token environment variable for Desktop learning, and improves download reliability and deep-linking, alongside a database migration and the removal of memories from the personal edition.

└──▷ **WHAT SHIPPED · 7 FEATURES** most completely described first ? what's the number?

01 **Configurable skill approval and sharing in Automatic Learning**

NEW 90

Admins can set Who approves skills learned from people's recordings to Admins or Each person, for their own Bots under Admin → Automatic learning. The personal edition now has Settings → Learning with Learn now and Approve new skills automatically controls, approving skill proposals shortly after each start and every 15 minutes. Skills approved by an individual reach only their own Bots until they press Ask to share and an admin approves the request under Admin → Automatic learning → Sharing requests.

Let individuals review and optionally share their own learned skills rather than routing all approvals through admins.

📍 In Admin → Automatic learning, set 'Who approves skills learned from people\'s recordings' to Each person, for their own Bots. Users then review pending skills under Settings → Learning and press Ask to share to request broader publication.

— Exact setting names and navigation paths given, full workflow explained v0.3.4

02 **Model token for Desktop Automatic Learning** NEW 60

OpenBot Desktop can now pass a model token to Automatic Learning via the new optional `OPENBOT_LEARNING_MODEL_TOKEN` environment variable set in `docker-compose.yml`.

— Env var and file named but no further mechanism described v0.3.4

03 **Desktop download sourcing and retry reliability** IMPROVED 60

OpenBot Desktop now prioritises `downloads.copilotkit.ai` over GitHub, ghcr.io and Docker Hub for container images, source archives, release lookups and back-end image pulls. Stalled GitHub downloads during install or update now retry up to six times with increasing back-off, up to roughly four minutes total, instead of giving up after three attempts.

— Named sources and retry numbers, but it's automatic background behavior v0.3.4

04 **Learning skill approvals schema migration** BREAKING 60

The migration `0083_learning_skill_approvals` advances the database to schema `0083_learning_skill_approvals` and creates the table `learning_skill_approvals`. Downgrading to v0.3.3 after an admin changes the approval routing setting, or after Approve new skills automatically is toggled in the personal edition, causes every conversation to fail on v0.3.3 because it cannot read the saved Learning settings; restore a pre-upgrade backup instead of downgrading.

— Named migration and table, with a concrete recovery instruction v0.3.4

THINNER COVERAGE BELOW

05 **Live recordings list in Teach a Bot** IMPROVED 50

Teach a Bot now lists your recordings, newest first with outcome status, and updates live when the Recorder sends a new one. This requires Intelligence 0.16 with recordings enabled.

— Behavior and version requirement named, no exact UI path given v0.3.4

06 **openbot:// deep links from notifications** NEW 40

OpenBot Desktop now registers `openbot://` links so clicking a system notification opens the relevant conversation directly, including from the Windows notification centre.

— Names the link scheme but little further mechanism v0.3.4

07 **Memories removed from personal edition** DEPRECATED 35

The personal edition no longer shows, uses or syncs memories or memory sources; existing memories remain in the database but are hidden from all Bots after upgrade.

— States the change but gives no migration path or reasoning v0.3.4

└──▷ BREAKING ON UPGRADE

! Downgrading to v0.3.3 after an admin changes 'Who approves skills learned from people's recordings', or after Approve new skills automatically is toggled in the personal edition, causes every conversation to fail on v0.3.3, which cannot read the saved Learning settings. Restore a pre-upgrade backup instead of downgrading.

! The personal edition no longer shows, uses or syncs memories or memory sources; existing memories remain in the database but are hidden from all Bots after upgrade.

! The migration `0083_learning_skill_approvals` advances the database to schema `0083_learning_skill_approvals` and creates the table `learning_skill_approvals`.

WAS THIS USEFUL?  

◆ Agent Memory

claude-mem

Feedback ⓘ v15.18.0 NOTES ↗ RANK

claude-mem gives coding agents persistent memory by capturing session activity, compressing it with a model, and injecting relevant context into later sessions.

> claude-mem's v15.18.0 release extends its memory loop to new agent surfaces — Gemini CLI and eve agents — and adds configurable model/provider routing for corpus queries and eve runs, plus a browser-based setup flow for new installs.

└──▷ WHAT SHIPPED · 4 FEATURES most completely described first ? what's the number?

01 eve agent memory loop integration NEW INTEGRATION NEW

90

eve agents now get the full claude-mem memory loop — session timeline context, per-turn recall, and recording — via `claude-mem/eve`, tested on eve 0.84.0 and 1.0.0-rc.9. Recording is on by default; pass `capture: false` to keep read-only behaviour while retaining session context, recall, search, and `remember`. Set `EVE_MODEL` to pick an exact model for eve agent runs, or set `ANTHROPIC_API_KEY` to call Claude directly instead of routing through the Vercel AI Gateway.

— Names module, flag, env vars, and tested eve versions v15.18.0

02 Corpus queries via OpenRouter/OpenAI providers NEW

85

Set `CLAUDE_MEM_CORPUS_AGENT=api` to route `query_corpus` calls through any OpenRouter or OpenAI-compatible provider; the default Claude Agent SDK path is unchanged if left unset.

— Exact env var and default documented, mechanism otherwise thin v15.18.0

03 Gemini CLI lifecycle capture NEW

80

claude-mem now supports Gemini CLI sessions, installed and managed via `npx claude-mem gemini install`, `npx claude-mem gemini status`, or `npx claude-mem gemini uninstall`, recording Gemini CLI activity to a cmem.ai account.

Set up Gemini CLI with native lifecycle memory capture so Gemini sessions are recorded to your cmem.ai account.

```
$ npx claude-mem gemini install
```

— Exact runnable subcommands, but capture mechanism unexplained v15.18.0

04 **Browser-based setup for standard install** IMPROVED 75

Standard install now opens a browser-based setup step for agents, integrations, model, and memory preferences. Agent shells can pass `--setup browser` when a person is present, or `--setup defer` to leave choices for the first My Work visit.

— Names exact flags but setup flow details are sparse v15.18.0

v15.17.1 NOTES ↗ STABLE

claude-mem's plugin cache sweep now respects open-session `.in_use` records, keeping captured sessions alive through updates; `npx claude-mem prune --dry-run` follows the same rule.

└──▷ GET THIS VERSION ▾

npm

```
$ npm install -g claude-mem@15.17.1
```

git

```
$ git clone --branch v15.17.1 https://github.com/thedotmack/claude-mem.git
# already have the repo? check out this version:
$ git checkout v15.17.1
```

signed build provenance · checked 12 Nov Wrong command?

WAS THIS USEFUL? 👍 👎

◆ Applied AI Agents

ZeroClaw

Feedback ⓘ v0.9.0 NOTES ↗ RANK

ZeroClaw runs personal AI assistants with configurable model providers, tools, memory, and messaging channels.

> ZeroClaw's v0.9.0 release adds OIDC-based multi-account dashboard sign-in with scoped per-account chat, CLI-only server builds, ZeroCode session lifecycle controls, and several messaging-channel and model-handling improvements, alongside a breaking config schema bump.

└──▷ WHAT SHIPPED · 9 FEATURES most completely described first ? what's the number?

01 OIDC sign-in and scoped dashboard chat for accounts NEW

95

Opt-in OIDC sign-in for the dashboard is configured via `[gateway.dashboard_sign_in]` with one or more `[oidc.<alias>]` provider blocks; access tokens stay in gateway memory and the browser holds only an `HttpOnly` session cookie. Non-administrator accounts get their own scoped dashboard chat — a signed-in account whose profile grants `sessions` can chat with the agents its grants allow, with conversations private to that account.

Add an OIDC provider so dashboard users can sign in through your own identity provider, with sessions scoped to their permission profile.

toml

```toml
[gateway.dashboard_sign_in]

[oidc.corp]
issuer = "https://sso.example.com"
client_id = "zeroclaw-dashboard"
client_secret = "s3cr3t"
```

— Full mechanism, config keys, and runnable example provided v0.9.0

02 ZeroCode session lifecycle controls NEW

85

Sign in without pairing by running `zerocode login` from an empty client directory — the daemon issues the client certificate after verifying the OIDC token and the account's grants. Resume a saved session by name or id with the new `/load-session` endpoint. Control how long orphaned remote sessions are kept before automatic closure with `wss.orphaned_session_timeout_secs` (default: 30 minutes).

— Named command, endpoint and config key, no example given v0.9.0

03 **CLI-only server install and build switching** NEW 75

ZeroClaw can be installed without the web dashboard using `install.sh --cli-only` or `install.sh --preset cli-only`, and switched back to the standard build later with `zeroclaw update --standard`.

Deploy a headless ZeroClaw on a server where the web dashboard is not needed, then switch to the standard build later if required.

```
$ curl -fsSL https://raw.githubusercontent.com/zeroclaw-
labs/zeroclaw/master/install.sh | sh -s -- --cli-only
# Later, to add the dashboard:
zeroclaw update --standard
```

— Exact install and switch-back commands given v0.9.0

04 **Custom unauthorized message for Telegram** NEW 75

Set a custom unauthorized-user notice for Telegram with `channels.telegram.<alias>.unauthorized_message`, which accepts the `{identity}` and `{bind_command}` placeholders.

Set a custom message for Telegram users who are not yet authorized, telling them how to request access.

```toml
[channels.telegram.default]
unauthorized_message = "Hi {identity}! You are not authorized.
Pair with: {bind_command}"
```

— Exact config key with example and placeholders v0.9.0

05 **Config schema version 4 migration required** BREAKING 65

The configuration schema advances to version 4; configs must be updated before upgrading by setting `schema_version = 4` at the top of `~/.zeroclaw/config.toml` and reviewing the migration notes in the release's Breaking Changes section.

— Exact config key and file path for required migration v0.9.0

THINNER COVERAGE BELOW

06 **Inbound-only Email channel mode** NEW 55

Configure the Email channel inbound-only (no SMTP) by leaving `smtp_host` and `from_address` unset or blank.

— Named config keys but no usage example v0.9.0

07 **Adaptive thinking format for newer Claude models** IMPROVED

55

Claude 4.7 and later models (Opus, Sonnet, Haiku, Fable, Mythos from version 4.7 on, including gateway-prefixed or suffixed names such as `-fast`) now receive adaptive thinking format automatically instead of the budget format, which those models previously rejected.

— Explains behavior change; no user action required v0.9.0

08 **File sending and receiving in Signal channel** NEW

45

The Signal channel now sends and receives files — photos, voice notes and documents are saved into the agent's workspace.

— Describes capability but no config or command v0.9.0

09 **Built-in docs skill for agents** NEW

40

A built-in `zeroclaw` docs skill lets an agent find documentation for the installed version.

— Brief description, no example or config shown v0.9.0

└──▷ **BREAKING ON UPGRADE**

! The configuration schema advances to version 4; configs must be updated before upgrading (set `schema_version = 4` at the top of `~/.zeroclaw/config.toml` and review the migration notes in the Breaking Changes section of the release).

WAS THIS USEFUL? 👍 👎

◆ AI Agent Communication

AG-UI

Feedback ⓘ release/2026-11-12 NOTES ↗ RANK

AG-UI: the Agent-User Interaction Protocol. Bring Agents into Frontend Applications.

> AG-UI's single release this window adds subagent activity tracking and recursive JSON Schema conversion for frontend tool parameters in the Claude SDKs, alongside three breaking changes tightening event payloads and template validation.

└──▷ WHAT SHIPPED · 4 FEATURES most completely described first ? what's the number?

01 Recursive schema conversion and key handling in @ag-ui/claude-agent-sdk

75

BREAKING

Frontend tool parameter schemas in `@ag-ui/claude-agent-sdk` are now converted from JSON Schema recursively, preserving nested object properties, array item schemas, required fields, enums, `const`, nullable/union types and descriptions. Nested objects also no longer strip undeclared keys during parsing. In `@ag-ui/claude-agent-sdk@0.0.9` this means parameter schemas carry nested structure instead of collapsing to `any[]` or open records, so agents relying on the previous loose shapes should be verified.

— Lists exact schema constructs preserved and migration note release/2026-11-12

02 threadId/runId removed from message and tool call events

67

BREAKING

In `ag-ui-claude-sdk@0.2.4`, `TEXT_MESSAGE_*` and `TOOL_CALL_*` events no longer include `threadId` / `runId`; consumers relying on those fields must read them from `RUN_STARTED` or other run-level events instead.

— Names exact fields and events plus the migration path release/2026-11-12

THINNER COVERAGE BELOW

03 **Subagent activity tracking in ag-ui-claude-sdk** NEW

59

Task tool calls in `ag-ui-claude-sdk` now open an `ActivitySnapshotEvent` / `ActivityDeltaEvent` activity so frontends can render dispatched subagents distinctly; activities close on teardown and on matching tool results.

— Names exact events and close conditions, no usage example `release/2026-11-12`

04 **A2UI template data path validation in ag_ui_adk** BREAKING

38

In `ag_ui_adk@0.13.0`, native A2UI template data paths are now validated; previously accepted invalid paths may be rejected.

— States the change but no example of valid/invalid paths `release/2026-11-12`

└──▷ **BREAKING ON UPGRADE**

! In `@ag-ui/claude-agent-sdk@0.0.9`, frontend tool parameter schemas now carry nested structure instead of collapsing to `any[]` or open records; verify agents relying on the previous loose shapes.

! In `ag-ui-claude-sdk@0.2.4`, `TEXT_MESSAGE_*` and `TOOL_CALL_*` events no longer include `threadId` / `runId`; consumers relying on those fields must read them from `RUN_STARTED` or other run-level events.

! In `ag_ui_adk@0.13.0`, native A2UI template data paths are now validated; previously accepted invalid paths may be rejected.

WAS THIS USEFUL? 👍 👎

DEPLOY

◆ ML Platforms & SDKs

Google Gemini Enterprise Agent Platform

Feedback ⓘ changelog · Nov 12, 2026 CHANGELOG ↗ RANK

Gemini Enterprise Agent Platform provides Google's managed model training, inference, evaluation, and agent deployment services, formerly offered under Vertex AI.

> Gemini Enterprise Agent Platform expanded long-running inference and agent governance, adding cross-cloud Kubernetes governance for Agent Gateway, Meta Muse models via the Responses API, and long-decode and latency controls for Gemini 4 Argon and Claude models.

└──▷ WHAT SHIPPED · 6 FEATURES most completely described first ? what's the number?

01 Cross-cloud Kubernetes governance via Agent Gateway NEW

85

Govern agents on CNCF-conformant Kubernetes clusters outside Google Cloud — including Amazon EKS, AKS, Red Hat OpenShift, and bare-metal — via Agent Gateway for cross-cloud (Limited Preview), routing LLM, MCP, and A2A traffic through local proxies that enforce access and Model Armor safety policies and send logs to Cloud Logging and Cloud Trace. Semantic governance policies can also be applied in a multi-project hub-and-spoke architecture by configuring a Resource Manager management boundary with the `AGENT_MANAGEMENT` capability so a single Agent Gateway and policy engine governs agents across all member projects.

— Mechanism and capability named in depth, no runnable command

changelog-20261112-9ed294bb

02 Standalone video upscaling in Gemini Omni 1.2 Flash NEW

80

Upscale a previously generated video (e.g. 360p to 1080p) without re-generating it from scratch, by calling the `predictLongRunning` endpoint with `"task": "upscale"` and a `targetResolution` field alongside the source video's `gcsUri`.

Upscale a previously generated 360p video to 1080p using Gemini Omni 1.2 Flash without re-generating it from scratch.

```
$ curl -X POST https://us-central1-
aiplatform.googleapis.com/v1/projects/<PROJECT>/locations/us-
central1/publishers/google/models/gemini-omni-1.2-
flash:predictLongRunning \
  -H 'Authorization: Bearer $(gcloud auth print-access-token)' \
  -H 'Content-Type: application/json' \
  -d '{
    "instances": [{
      "task": "upscale",
      "video": {"gcsUri": "gs://<BUCKET>/input-360p.mp4"},
      "targetResolution": "1080p"
    }]
  }'
```

— Full curl command with endpoint and fields given `changelog-20261112-9ed294bb`

03 **Meta Muse models via Responses API** NEW 75

Access three Meta Muse models in Preview — `meta/muse-wonder-1.0`, `meta/muse-ember-1.0`, and `meta/muse-spark-1.3` — through the OpenAI-compatible Responses API on the global endpoint, supporting reasoning, function calling, structured output, and a 1M-token context window.

— Named models and capabilities, no endpoint call shown `changelog-20261112-9ed294bb`

04 **Long-running decodes in Interactions API** NEW 72

Use `background=true` for server-managed background execution or `store=true` / `store=false` for custom continuation loops when running Interactions API long decodes up to 1,048,576 tokens for Gemini 4 Argon.

— Flags and token limit named, no full runnable example `changelog-20261112-9ed294bb`

05 **Fast-speed mode for Claude predictions** NEW 63

Set `"speed": "fast"` in online pay-as-you-go prediction requests for supported Anthropic Claude models (starting with `claude-fable-5-2`) to accelerate output token generation for latency-sensitive agentic workflows.

— Named field and model, no request example shown `changelog-20261112-9ed294bb`

THINNER COVERAGE BELOW

06 **Gemini 4 Carbon frontier model for CodeMender** NEW 50

Run a CodeMender session with `cm fix --model gemini-4-carbon` to use the platform's latest frontier model for complex code-repair tasks.

Run a CodeMender session using Gemini 4 Carbon to get the platform's latest frontier model for complex code-repair tasks.

```
$ cm fix --model gemini-4-carbon
```

— Only a command example, no description of the model itself `changelog-20261112-9ed294bb`

WAS THIS USEFUL? 👍 👎

Hugging Face OpenEnv

Feedback ⓘ v0.13.0 NOTES ↗ RANK

OpenEnv is an interface library for RL post training with environments.

> OpenEnv v0.13.0 adds AgentEnv bundle importing, new runtime containment and isolation grading metrics, Harbor SFT rollout export for fine-tuning, and tighter validation of Hub Space sources, alongside a new OR-Tools environment.

└──▷ **WHAT SHIPPED · 5 FEATURES** most completely described first ? what's the number?

01 **AgentEnv bundle import command** NEW 55

OpenEnv can now import AgentEnv bundles via the new `openenv import` command.

— Names exact command but no further mechanism given v0.13.0

02 **Runtime containment and isolation grading** NEW 50

OpenEnv now grades `runtime.host_containment` from measured isolation, `runtime.resource_bounds` from measured limits, and `runtime.episode_isolation` and `runtime.oracle_containment`.

— Names four metrics but not how grading is used v0.13.0

03 **Harbor SFT rollout export** NEW 40

Harbor now exports rollouts as SFT rows for supervised fine-tuning workflows.

— States output format but no command or file path v0.13.0

04 **Hub Space source validation and HF runtime gating** NEW 25

OpenEnv now validates Hub Space sources and gates unverified HF runtime.

— Brief mention with no mechanism or config shown v0.13.0

05 **OR-Tools environment added** NEW 15

OpenEnv adds an OR-Tools environment, per the release summary.

— Only named in summary, no detail provided v0.13.0

WAS THIS USEFUL? 👍 👎

Google Gen AI JavaScript SDK

Feedback ⓘ v3.2.0

NOTES ↗ RANK

The Google Gen AI JavaScript SDK provides TypeScript and JavaScript interfaces to Gemini APIs and live model sessions.

> Google Gen AI JavaScript SDK v3.2.0 adds a new Decisions API and expands self-hosted BYOS environment configuration with new fields.

└──▷ **WHAT SHIPPED · 2 FEATURES** most completely described first ? what's the number?

01 **BYOS environment configuration fields** NEW 60

Self-hosted BYOS environments can now be configured using the new `system_instruction`, `custom_config`, and `environment_key` fields in the SDK and CLI.

— Names config fields but not their effects. v3.2.0

THINNER COVERAGE BELOW

02 **Decisions API** NEW 55

The SDK adds a new Decisions API, callable via `client.decisions.create`.

— Names the method but not its behavior or parameters. v3.2.0

WAS THIS USEFUL? 👍 👎

Google Gen AI Python SDK

Feedback ⓘ v3.2.0 NOTES ↗

RANK

The Google Gen AI Python SDK provides Python interfaces to Gemini model generation, streaming, tools, and embeddings.

> Google Gen AI Python SDK v3.2.0 introduces a new Decisions API alongside configuration fields for self-hosted BYOS environments.

└──▷ **WHAT SHIPPED · 2 FEATURES** most completely described first ? what's the number?

01 **BYOS environment configuration fields** NEW 65

Configure self-hosted BYOS environments with the new `system_instruction`, `custom_config`, and `environment_key` fields in the environments module.

— Names exact config fields but lacks usage mechanism v3.2.0

02 **Decisions API for structured workflows** NEW 60

Call `client.decisions.create` to use the new Decisions API for structured decision-making workflows.

— Names exact method but no detail on mechanism or outputs v3.2.0

WAS THIS USEFUL? 👍 👎

siliconscope

Feedback ⓘ v4.8.2 NOTES ↗ RANK

Siliconscope sudoless Apple Silicon system monitor (native SwiftUI GUI) with ANE / Media Engine / memory-bandwidth tracking

> SiliconScope added an opt-in setting to keep chart measurement running continuously even when its window is hidden, minimized, or closed.

└──▷ **WHAT SHIPPED · 1 FEATURE** most completely described first ? what's the number?

01 **Continuous measurement while window hidden** NEW 69

A new setting, Settings → Keep measuring while the window is hidden, keeps charts measuring continuously when the window is hidden, minimized, or closed, avoiding gaps during long unattended monitoring sessions. It is off by default and uses about one percent more of a CPU core when enabled.

Keep all charts gap-free during long unattended monitoring sessions where the window may be hidden or minimized.

📍 In the app, go to Settings → Keep measuring while the window is hidden and toggle it on.

— Exact setting path and CPU cost given, but no deeper mechanism. v4.8.2

WAS THIS USEFUL? 👍 👎

◆ Inference Providers & GPU Clouds

Modal

Feedback ⓘ 1.6.4 (2026-11-11) NOTES ↗ RANK

Modal is a cloud infrastructure for teams that develop, train, and serve AI applications at scale.

> Modal's 1.6.4 release adds no-code deployment of autoscaling servers straight from container images, a full set of Image management CLI commands, workspace token lifecycle controls, new stats APIs covering functions, servers and sandboxes, and smaller additions for endpoint model discovery and class-level function lookups.

└──▷ **WHAT SHIPPED · 6 FEATURES** most completely described first ? what's the number?

01 **Workspace token disabling and rotation** NEW 88

Disable workspace API tokens without hard-deleting them via the new Workspace.member_tokens.disable() method and `modal workspace member-tokens disable` CLI, including a `--all` flag to disable every token at once; rotate the active profile token with `modal token rotate`, which creates a replacement and deletes the current one.

Quickly lock down a compromised workspace by disabling every member token at once, then re-enable specific ones after investigation.

```
$ modal workspace member-tokens disable --all
```

— Exact commands, flag, and method named with mechanism 1.6.4 (2026-11-11)

02 **No-code server deployment from images** NEW 85

The new `modal server deploy` CLI deploys an autoscaling server directly from a named Modal Image or a third-party registry URI, requiring only that the image start an HTTP server on a bound port.

Deploy a containerised HTTP service from a third-party registry image without writing any Python, as a no-code Modal deployment.

```
$ modal server deploy registry.example.com/myapp:latest
```

— Exact command, requirement, and runnable example given 1.6.4 (2026-11-11)

03 **Modal Image build, pull, and delete commands** NEW 81

Build a local Dockerfile into a named Modal Image for reuse across apps with the new `modal image build` CLI; pull an image from a third-party registry and publish it as a named Modal Image with `modal image pull`; and delete a named Modal Image association (without removing the underlying image) with `modal image names delete`.

Build a local Dockerfile into a named Modal Image so other apps can reference it without rebuilding.

```
$ modal image build ./Dockerfile my-org/base-image:latest
```

— Three CLI subcommands named with exact usage `1.6.4 (2026-11-11)`

04 **App, Function, Server, and Sandbox stats APIs** NEW

70

Report Function, Server, and Sandbox stats for an app with the new App.stats() method and `modal app stats` CLI; query Sandbox-specific stats including costs with App.sandboxes.stats(), with aggregation support based on Sandbox tags.

— Named methods and CLI but no usage example given `1.6.4 (2026-11-11)`

THINNER COVERAGE BELOW

05 **Class-level Function lookup via from_id** NEW

55

Pass a `fu-` class ID to Function.from_id() to get an object supporting observability and autoscaling APIs for a Modal Cls.

— Names method and ID prefix but no usage example `1.6.4 (2026-11-11)`

06 **List models servable via Modal Endpoints** NEW

45

List models available to serve with Modal Endpoints using the new `modal endpoint library` CLI.

— Single command named, no further detail given `1.6.4 (2026-11-11)`

WAS THIS USEFUL? 👍 👎

Oracle OCI Generative AI

Feedback ⓘ

Import DeepSeek V4.1 Flash 1017 into OCI Generative AI (2026-11-05) BLOG ↗

OCI Generative AI is Oracle's Enterprise AI Platform for building, deploying, and governing AI applications at scale.

OCI Generative AI now supports importing DeepSeek-V4.1-Flash-1017 for multimodal image-and-text inference on dedicated GPU cluster shapes.

NEW SUPPORTED PLATFORM Import `deepseek-ai/DeepSeek-V4.1-Flash-1017` with model capability `IMAGE_TEXT_TO_TEXT` and deploy on dedicated AI cluster shapes `H100_X8`, `H200_X4`, or `B200_X4`.

WAS THIS USEFUL? 👍 👎

DATA

◆ Vector DBs & RAG

ThHanke Ontosphere

Feedback ⓘ v1.8.0 NOTES ↗ RANK

Ontosphere is a browser-based RDF/ontology knowledge graph editor — load RDF from files, URLs or SPARQL endpoints; author nodes and edges on the canvas; run OWL-RL reasoning; apply multi-algorithm layout and clustering. AI-native via MCP tool surface and relay bridge.

> Ontosphere's v1.8.0 release substantially speeds up reasoning and canvas rendering for large graphs, adds per-stage timing and better inconsistency handling to `explainDiagnostics`, and introduces startup gating for headless and MCP automation — alongside breaking changes to SHACL shapes loading, justification output, and IRI text formatting.

└──▷ **WHAT SHIPPED · 8 FEATURES** most completely described first ? what's the number?

01 Per-stage timing and inconsistency handling in explainDiagnostics

IMPROVED 87

`explainDiagnostics` now accepts `{ includeTimings: true }` to return per-stage timing breakdowns for profiling reasoning pipelines. It skips `unsatisfiableClasses` for inconsistent ontologies, signaling this via `unsatisfiableClassesSkipped: "ontology-inconsistent"`, and now completes on large inconsistent ontologies in 128s instead of 513s.

— Names exact flag and field with concrete before/after timing v1.8.0

02 Startup gating for headless runs and MCP calls

BREAKING 86

Skip the PyodideSemanticWorkflow catalog load on headless or evaluation runs with the `?workflowCatalog=off` URL parameter, then check startup completion programmatically via window.__startupComplete() before calling MCP tools. MCP tool calls now return `success: false` with a still loading message until window.__startupComplete() is `true`; scripts and agents that call a tool immediately after page open must wait for that flag or retry.

Run Ontosphere headlessly without blocking on the workflow catalog load, then wait for startup before invoking any MCP tool.

📍 `https://your-ontosphere-instance/?workflowCatalog=off&rdfUrl=https://example.org/ontology.ttl` // In your automation script, poll before calling tools: await waitUntil(() => window.__startupComplete() === true); await runReasoning();

— Names exact URL param and function with a runnable polling example v1.8.0

03 **SHACL shapes loading precedence order** BREAKING 83

Shapes loading now follows a precedence order: `?shaclShapes=` URL parameter, then the configured URL, then the built-in ontology-quality shapes. An empty SHACL shapes URL now means 'use the built-in ontology-quality shapes' instead of 'do not load shapes'; use the SHACL toggle to disable SHACL entirely.

— Names exact parameter and precedence order with disable instructions v1.8.0

04 **Removal of laconic justifications and other explainDiagnostics changes**

76

BREAKING

`explainDiagnostics` no longer returns laconic explanations — `laconicJustifications` and the Precise culprit lines in `repairBrief` are removed; read `justifications` instead. Justifications no longer include declarations, the number of justifications can exceed `maxJustifications`, and weakenings are offered only for a single independent clash.

— Lists exact removed fields and new justification behavior v1.8.0

05 **Performance improvements for large graph reasoning and rendering**

IMPROVED 75

Reasoning on LUBM-1 drops from 126s to 11.4–11.7s with inferred triples reaching the canvas in chunks, canvas element load time for 8,000 individuals drops from 31s to 19s, and renderer memory per individual drops from about 0.34 MB to 0.26 MB.

— Concrete before/after numbers across three metrics but no user action v1.8.0

06 **Inferred triple count changes from reasoner upgrade** BREAKING 73

Inferred triple counts change when upgrading from `rdf-reasoner-konclude` 0.8.0 to 0.9.3 — Pizza goes from 137 to 170 inferred triples; re-measure any figure published with v1.7.x.

— Names exact version upgrade and concrete triple counts v1.8.0

07 **IRI naming format change in text results** BREAKING 62

Text results now name IRIs as prefixed names or full IRIs (e.g. `inc:frank`) instead of bare local names (e.g. `frank`); agents that parse these texts must adapt.

— Gives example format change but no migration tool v1.8.0

THINNER COVERAGE BELOW

08 **Relation labels shown on selection for large canvases** NEW 50

On large canvases with more than 300 links, relation labels now show only on selection, reducing visual noise while keeping links clickable.

— Names threshold but no toggle or config key given v1.8.0

└──▷ **BREAKING ON UPGRADE**

! Inferred triple counts change when upgrading from rdf-reasoner-konclude 0.8.0 to 0.9.3 — Pizza goes from 137 to 170 inferred triples; re-measure any figure published with v1.7.x.

! MCP tool calls now return `success: false` with a still loading message until window.__startupComplete() is `true`; scripts and agents that call a tool immediately after page open must wait for that flag or retry.

! An empty SHACL shapes URL now means 'use the built-in ontology-quality shapes' instead of 'do not load shapes'; use the SHACL toggle to disable SHACL entirely.

! `explainDiagnostics` no longer returns laconic explanations — `laconicJustifications` and the Precise culprit lines in `repairBrief` are removed; read `justifications` instead.

! `explainDiagnostics` justifications no longer include declarations, the number of justifications can exceed `maxJustifications`, and weakenings are offered only for a single independent clash.

! Text results now name IRIs as prefixed names or full IRIs (e.g. `inc:frank`) instead of bare local names (e.g. `frank`); agents that parse these texts must adapt.

WAS THIS USEFUL?  

helix-db

Feedback ⓘ v3.7.0 NOTES ↗ RANK

HelixDB is an OLTP graph database with native vector and full-text search built in Rust on Object Storage.

> helix-db v3.7.0 adds per-project telemetry attribution for open-source deployments and improves storage durability timeouts, SDK ergonomics, and write throughput via per-writer threading.

└──▷ **WHAT SHIPPED · 4 FEATURES** most completely described first ? what's the number?

01 **30-second durable request timeout** IMPROVED 40

helix-db now times out durable requests waiting for storage at 30 seconds.

— Names a concrete limit but no config to change it v3.7.0

02 **Re-exported embedded cache config types in Rust SDK** IMPROVED 40

helix-db now re-exports embedded cache config types from the Rust SDK, making them accessible without additional imports.

— Names SDK surface but no exact type names given v3.7.0

03 **Per-project OSS telemetry attribution** NEW 35

helix-db now attributes OSS telemetry to individual projects and deployments, counting every query executed.

— Describes what it does but no config keys or mechanism v3.7.0

04 **Dedicated thread per writer for commits** IMPROVED 35

helix-db now processes each writer's commits on a dedicated thread, increasing write throughput.

— States mechanism and effect but no benchmarks v3.7.0

WAS THIS USEFUL? 👍 👎

ralforion OrionBelt Ontology Builder

Feedback ⓘ

v1.31.2 NOTES ↗ RANK

OrionBelt Ontology Builder is a browser-based ontology workbench for OWL ontologies and SKOS vocabularies. Streamlit + rdflib, no Java, no Protégé.

> OrionBelt Ontology Builder sped up large-file parsing and made the SKOS browsing UI update incrementally instead of rebuilding from scratch.

└──▷ **WHAT SHIPPED · 2 FEATURES** most completely described first ? what's the number?

01 **Incremental SKOS concept list updates** IMPROVED 56

The SKOS concept list, per-scheme list, and concept hierarchy now update incrementally on edit instead of rebuilding in full, dropping post-edit time from 147 ms to 1 ms with 5,000 concepts.

— Clear before/after numbers but no UI path or command given v1.31.2

02 **Faster Turtle/N-Triples parsing via dual Oxigraph/rdflib path**

IMPROVED 55

Large Turtle and N-Triples files now parse faster using a dual Oxigraph/rdflib path, cutting a 213,000-triple load from 3.5s to 2.1s.

— Names mechanism and numbers but no config or flag to invoke it v1.31.2

WAS THIS USEFUL? 👍 👎

Hugging Face Sentence Transformers

Feedback ⓘ

v6.3.0 NOTES ↗ RANK

Sentence Transformers trains and runs embedding, retrieval, and reranking models, and is the reference implementation behind most open embedding models.

> Sentence Transformers' v6.3.0 release focuses on multi-vector (ColBERT-style) models, adding loss wrappers to preserve retrieval quality under token pooling, hierarchical pooling controls for document vector budgets, optional acceleration kernels, native AWS Neuron inference, and faster clustering, alongside a breaking change to how built-in multi-vector losses compute similarity.

└──▷ **WHAT SHIPPED · 7 FEATURES** most completely described first ? what's the number?

01 Multi-vector training losses for retrieval quality NEW

95

`MultiVectorTokenPoolingLoss` wraps any retrieval loss to reuse encoder embeddings and train on both unpooled and pooled objectives in a single pass, while `MultiVectorMatryoshkaLoss` wraps a retrieval loss with `matryoshka_dims=[128, 64, 32]` and `matryoshka_weights=[...]` so shorter vectors can be used at inference via `truncate_dim=64` on `encode_query` and `encode_document`. This lets multi-vector (ColBERT-style) models retain retrieval quality after pooling or dimensionality reduction.

Train a ColBERT-style model so its document embeddings stay useful after 4x token pooling, without an extra encoder pass.

```python
from sentence_transformers import MultiVectorEncoder
from sentence_transformers.multi_vector_encoder.losses import (
    CachedMultiVectorMultipleNegativesRankingLoss,
    MultiVectorTokenPoolingLoss,
)
from sentence_transformers.multi_vector_encoder.modules import
HierarchicalTokenPooling

model = MultiVectorEncoder('answerdotai/ModernBERT-base')
pooling = HierarchicalTokenPooling(pool_factor=4)
base_loss = CachedMultiVectorMultipleNegativesRankingLoss(model,
mini_batch_size=64)
loss = MultiVectorTokenPoolingLoss(
    model,
    base_loss,
    token_poolings=[None, pooling],
    weights=[0.5, 0.5],
)
# pass loss to MultiVectorEncoderTrainer
```

— Named loss classes, params and encode flags with a runnable example v6.3.0

02 **Hierarchical token pooling for document vector budgets** NEW

90

HierarchicalTokenPooling(pool_factor=4), HierarchicalTokenPooling(max_tokens=16), or HierarchicalTokenPooling(distance_threshold=...) can be passed to `encode_document` or as `token_pooling=` on the `MultiVectorEncoder` constructor to control document vector budgets. Adaptive threshold calibration from representative documents is available via HierarchicalTokenPooling.calibrate_distance_threshold(embeddings, target_tokens=8), which returns `None` when no pooling is needed.

— Named config class and methods but no end-to-end example v6.3.0

03 **Optional kernels for faster inference** NEW

90

Setting `use_kernels=True` on any of the four model types accelerates transformer backbones, MaxSim and MeanMaxSim scoring, and Ward-clustering-based token pooling; dependencies install via `pip install 'sentence-transformers[kernels]'`.

Speed up MaxSim scoring and Ward-clustering token pooling on an RTX GPU with a single constructor flag.

```python
from sentence_transformers import MultiVectorEncoder
from sentence_transformers.multi_vector_encoder.modules import
HierarchicalTokenPooling

model = MultiVectorEncoder('lightonai/mLateOn', use_kernels=True)
queries = model.encode_query(['How can I reduce retrieval
storage?'])
documents = model.encode_document(
    ['Token pooling merges similar document vectors before
indexing.'],
    token_pooling=HierarchicalTokenPooling(pool_factor=2),
)
scores = model.similarity(queries, documents)
```

— Exact flag and install command with runnable example v6.3.0

04 **Similarity function change for built-in multi-vector losses**

90

BREAKING

The four built-in contrastive and distillation losses for multi-vector models now follow the model's similarity function even with kernels disabled, so MeanMaxSim models use mean scores instead of MaxSim sums. To retain the old MaxSim behavior, set `similarity_fct` to `colbert_scores` for ranking losses, `colbert_scores_pairwise` for margin MSE, or `colbert_kd_scores` for KL distillation.

— Exact before/after behavior with named config values to restore it v6.3.0

05 **Native AWS Neuron inference** NEW 80

Dense, cross-encoder, and multi-vector inference can now run on AWS Inferentia/Trainium by setting `device='neuron'` on `SentenceTransformer` or `MultiVectorEncoder`, using the existing torch backend after installing TorchNeuron.

Run a multi-vector model on AWS Inferentia/Trainium cores without changing any encode calls.

```python
from sentence_transformers import SentenceTransformer

model = SentenceTransformer('sentence-transformers/all-MiniLM-L6-v2', device='neuron')
embeddings = model.encode(['Native inference on AWS Neuron.'])
```

— Named device flag and hardware with a working code example v6.3.0

THINNER COVERAGE BELOW

06 **Faster community detection and clustering** IMPROVED 50

Community detection now reduces GPU memory use and speeds up CPU clustering through a shared batched implementation, and speeds up community detection and hard-negative mining on Apple Silicon by reducing MPS device-synchronization overhead.

— Mechanism described but no flag or command to invoke it v6.3.0

07 **Reduced data-preparation overhead** IMPROVED 40

Encoding, prediction, mining, and selected sampler and evaluator workflows now reduce data-preparation overhead through batch conversion of Hugging Face Dataset columns.

— General internal improvement without a named surface to act on v6.3.0

└──▷ **BREAKING ON UPGRADE**

! The four built-in contrastive and distillation losses for multi-vector models now follow the model's similarity function even with kernels disabled, so MeanMaxSim models use mean scores instead of MaxSim sums. To retain the old MaxSim behavior, set `similarity_fct` to `colbert_scores` for ranking losses, `colbert_scores_pairwise` for margin MSE, or `colbert_kd_scores` for KL distillation.

WAS THIS USEFUL? 👍 👎

EVALUATE

◆ AI Observability & Evals

LangChain LangSmith

Feedback ⓘ updated · Nov 12, 2026

NOTES ↗ RANK

LangSmith provides tracing, evaluation, and deployment tools for LLM applications.

> LangSmith shipped filtering and permission improvements to its workspace and tagging UI, added backward compatibility for older trace links, moved thread automations to SmithDB query configuration, and began deprecating the Turns view in favor of Messages/Details.

└──▷ **WHAT SHIPPED · 3 FEATURES** most completely described first ? what's the number?

01 **API Keys page full-list search** IMPROVED 45

The API Keys page now filters keys across the full paginated list by `key`, `description`, and `workspace`.

— Names filter fields but no further mechanism detail snapshot-20261112

02 **Tag-on-create permission for resources** IMPROVED 35

Users with the tag-on-create permission for datasets, projects, or prompts can now see and apply tags when creating those resources, instead of only workspace managers.

— Describes permission change but no config/UI path given snapshot-20261112

03 **Backward compatibility for older trace links** IMPROVED 25

LangSmith now recognizes older trace links containing timestamp-prefixed run and trace IDs.

— Bare statement with no mechanism or action for reader snapshot-20261112

└──▷ **BREAKING ON UPGRADE**

! The Turns view in thread detail is deprecated and will be removed from LangSmith Cloud on October 31, 2026, and from self-hosted in v17; use the Messages or Details view instead.

! Thread annotation-queue and webhook automations now require SmithDB queries, which may break existing automations that used run-filter-based configurations.

WAS THIS USEFUL? 👍 👎

Superlog Labs Superlog Feedback ⓘ

changes since 2026-10-13 CODE ↗

Superlog monitors AI agent telemetry with OpenTelemetry and groups signals into incidents for observability.

Superlog adds Linear team selection per project, ClickHouse insert pacing controls, and a monthly spend limit for orgs

└──▷ TRY IT

Retrieve the Linear teams connected to a project so you can display or validate team choices before writing a default.

```
$ curl -H 'Authorization: Bearer <token>'
https://api.superlog.example/api/projects/<projectId>/linear/teams
```

Set a project's default Linear ticket team, or clear it, as part of an automated workspace setup script.

```
$ curl -X PATCH -H 'Authorization: Bearer <token>' -H 'Content-Type:
application/json' -d '{"linearDefaultTeamId": "<teamId>"}'
https://api.superlog.example/api/projects/<projectId>/automation
```

Get the Linear teams available to a project via the new `GET /api/projects/:projectId/linear/teams` endpoint, which returns `{ installed: false, teams: [] }` when no Linear connection exists and 502 when Linear is unreachable.

Control ClickHouse insert pacing with the `INGEST_CLICKHOUSE_MIN_INSERT_INTERVAL_MS` environment variable, which sets a minimum delay between inserts into the same table per worker so faster inserts grow batch size instead of producing more parts.

Disable HTTP keep-alive on direct ClickHouse connections with `INGEST_CLICKHOUSE_KEEP_ALIVE=false` to spread inserts across replicas evenly instead of accumulating on the slowest one.

Let orgs set a monthly spend limit that freezes billable feature overages once the estimated bill reaches the ceiling, while still allowing consumption of included allowances.

Pick a default Linear ticket team from a new `Default team` picker in the `File Linear ticket` step in Settings, scoped to the project being edited.

WAS THIS USEFUL?  

Braintrust

Feedback ⓘ updated · Nov 12, 2026 NOTES ↗

Braintrust provides evaluation, tracing, and improvement workflows for AI applications.

Braintrust adds pattern rules for Loop, in-browser thread translation, `--parent` trace continuation in the CLI, and an optional telemetry argument for `bt mcp run`.

Tune Loop's scheduled Patterns runs with pattern rules that narrow scope to specific traces via a SQL filter and list problems for Loop to watch for.

Read non-English threads in English in the Thread layout using the browser's built-in translation API in desktop Chrome or Edge, consuming no model credits.

Continue a trace across separate runs by passing `--parent` to `bt trace run` with the value returned by an SDK span's export(), attaching the coding-agent session below that span instead of starting a new trace.

Make MCP server calls without supplying a telemetry argument — `bt mcp run` now treats the telemetry argument as optional, so callers pass only the tool's own arguments.

NEW SUPPORTED PLATFORM Create service tokens without storing the key in Terraform state or plan files using the new ephemeral `braintrust_service_token` resource.

+3 more

NEW INTEGRATION OpenCode 2 tracing

NEW INTEGRATION MCP server tracing for fastmcp and mcp-handler (TypeScript SDK)

UPDATED INTEGRATION TypeScript SDK tracing for Google ADK, Cloudflare Agents, Mistral, LangGraph, LangSmith, Anthropic, and Gemini

NEW INTEGRATION Firebase Genkit flow tracing (Go SDK)

UPDATED INTEGRATION Anthropic Java SDK 3.x async client support

└──▷ BREAKING ON UPGRADE

! The Terraform provider now requires Terraform or OpenTofu 1.11.1 or later.

! `braintrust_ai_secret` now requires `secret_version` alongside `secret`, and `braintrust_env_var` now requires `value_version` alongside `value`; changing only the secret or value no longer updates it, and references to `secret` or `value` return `null`.

! The `braintrust_org_automation` Terraform resource is removed; delete its blocks from configuration and run `terraform state rm` on each address before running `terraform plan`.

WAS THIS USEFUL? 👍 👎

GOVERN

◆ Model & Agent Protection

julius

Feedback ⓘ v1.7.3 NOTES ↗ LATEST STABLE

Julius is a simple LLM service identification - translate IP:Port to Ollama, vLLM, LiteLLM, or 60+ other AI services in seconds

└──▷ GET THIS VERSION ▾

Go

```
$ go install github.com/praetorian-inc/julius/cmd/julius@v1.7.3
```

git

```
$ git clone --branch v1.7.3 https://github.com/praetorian-
inc/julius.git
# already have the repo? check out this version:
$ git checkout v1.7.3
```

Go module path matches the repo · checked 12 Nov Wrong command?

UPDATED DETECTION MCP probe updated to 2026-07-28 spec

NEW DETECTION Hermes Agent API server detection probe

▾ 1 more release in this issue · 2026-11-11 → 2026-11-12

v1.7.0 NOTES ↗ STABLE

└──▷ GET THIS VERSION ▾

Go

```
$ go install github.com/praetorian-inc/julius/cmd/julius@v1.7.0
```

git

```
$ git clone --branch v1.7.0 https://github.com/praetorian-
inc/julius.git
# already have the repo? check out this version:
$ git checkout v1.7.0
```

Go module path matches the repo · checked 12 Nov Wrong command?

WAS THIS USEFUL?

◆ AI Red Teaming & Testing

AI-Infra-Guard

Feedback ⓘ v4.6.10 NOTES ↗ RANK

AI-Infra-Guard is an AI red teaming platform that scans AI infrastructure for CVE vulnerabilities, MCP/Agent skill risks, jailbreak weaknesses, and API relay security issues.

> AI-Infra-Guard's single release in this window adds a local generative AI safety compliance self-check.

└──▷ **WHAT SHIPPED · 1 FEATURE** most completely described first ? what's the number?

01 **Local generative AI safety self-check** NEW 23

AI-Infra-Guard's compliance module now includes a local generative AI safety self-check.

— No mechanism, scope, or config details given v4.6.10

WAS THIS USEFUL?

◆ AI Security Posture & Discovery

Mend.io Mend AI

Feedback ⓘ (03-November-2026) NOTES ↗

Mend AI is an AI security service that discovers AI components and helps teams govern their use in software applications.

Mend AI adds Conversational System Prompts and AI Inventory reports; all reports now support organization-scope generation.

NEW DETECTION Mend AI now includes a Conversational System Prompts report that locates the system prompts behind chat-based AI applications and shows exactly where each is defined in code, so security teams can focus review on user-facing AI.

UPDATED DETECTION Mend AI now includes an AI Inventory report with one row per detected AI entity, covering models, system prompts, agents, agent configurations, tools, and MCP servers.

All Mend AI reports can now be generated at the organization scope, in addition to the existing application and project scopes.

└──▷ BREAKING ON UPGRADE

! The previous AI Inventory report, which covered AI models only, is now renamed Models Inventory .

WAS THIS USEFUL? 👍 👎

AI MODELS

◆ Frontier Models

Google Gemini API

Feedback ⓘ updated · Nov 12, 2026

NOTES ↗ RANK

Gemini API provides Gemini models for generating text, images, and other multimodal content in applications.

> Gemini API introduced a dedicated Decisions API for fast, calibrated classification-style outputs with zero generated tokens.

└──▷ **WHAT SHIPPED · 1 FEATURE** most completely described first ? what's the number?

01 **Decisions API with gemini-decision-1 model** NEW 55

A new `gemini-decision-1` model and Decisions API let developers make fast, calibrated binary, categorical, and rubric decisions across text, images, audio, video, and PDFs, returning results with zero output tokens.

— Names model and capability but no endpoint or usage example snapshot-20261112

WAS THIS USEFUL? 👍 👎

Anthropic

Feedback ⓘ updated · Nov 12, 2026 NOTES ↗

The Claude Developer Platform is the API for building applications on Anthropic's Claude models that generate, analyze, and automate work.

Claude Fable 5.2 and Claude Mythos 5.2 now share thinking blocks with each other; the API drops those blocks on any model switch outside this pair.

NEW SUPPORTED PLATFORM Claude Fable 5.2 and Claude Mythos 5.2 now read each other's thinking blocks in a conversation, with the API dropping those blocks when the conversation moves to any model outside this pair (including Claude Opus 5.5 or Claude Fable 5.1).

UPDATED SUPPORTED PLATFORM Dreams (research preview) now supports Claude Fable 5.2.

└─▷ BREAKING ON UPGRADE

! Thinking blocks are dropped by the API when a conversation moves from the Claude Fable 5.2 / Claude Mythos 5.2 pair to any other model, including Claude Opus 5.5 or Claude Fable 5.1.

WAS THIS USEFUL? 👍 👎

Unbiased Pareto 26.10

Feedback ⓘ

new snapshot · Nov 12, 2026 MODEL ↗

Pareto 26.10 is a multimodal model from Unbiased.

New snapshot 20260929 listed on OpenRouter (unbiased/pareto-26.10-preview), first seen 12 Nov 2026.

WAS THIS USEFUL? 👍 👎

Mancer Weaver (alpha)

Feedback ⓘ price cut · Nov 11, 2026

MODEL ↗

Weaver (alpha) is a chat model from Mancer.

Price cut on OpenRouter (mancer/weaver): $0.40 → $0.30 in, $0.75 → $0.50 out per million tokens (-33%), first seen 11 Nov 2026.

WAS THIS USEFUL? 👍 👎

◆ Open-Weight Models

Z.ai GLM 5.2

Feedback ⓘ price increase · Nov 11, 2026 MODEL ↗

GLM 5.2 is a chat model from Z.ai.

Price increase on OpenRouter (z-ai/glm-5.2): $0.006 → $0.007 in, $6.80 → $6.80 out per million tokens (+17%), first seen 11 Nov 2026.

WAS THIS USEFUL? 👍 👎

MiniMax M2.5

Feedback ⓘ context change · Nov 11, 2026 MODEL ↗

MiniMax M2.5 is a reasoning model from MiniMax.

Context window shrank from 205K to 200K tokens on OpenRouter (minimax/minimax-m2.5), first seen 11 Nov 2026.

WAS THIS USEFUL? 👍 👎

NVIDIA Nemotron 3 Super

Feedback ⓘ

context change · Nov 11, 2026 MODEL ↗

Nemotron 3 Super is a chat model from NVIDIA.

Context window shrank from 1049K to 262K tokens on OpenRouter (nvidia/nemotron-3-super-120b-a12b), first seen 11 Nov 2026.

WAS THIS USEFUL? 👍 👎

undi95 ReMM SLERP 13B

Feedback ⓘ price cut · Nov 11, 2026

MODEL ↗

ReMM SLERP 13B is a chat model from undi95.

Price cut on OpenRouter (undi95/remm-slerp-l2-13b): $0.35 → $0.30 in, $0.65 → $0.50 out per million tokens (-23%), first seen 11 Nov 2026.

WAS THIS USEFUL? 👍 👎

// END OF TRANSMISSION

The AI Toolchain · curated daily · November 12, 2026
Releases & features from the open-source and commercial AI tools that practitioners actually run.

▷ Subscribe Feedback on this issue